ISSN 2007-9737

# Disinformation and Fake News: The Infodemic in the Global Health Field

Jairo Eduardo Márquez-Díaz*

Universidad de Cundinamarca,
Facultad de Ingeniería, Chía,
Colombia

jemarquez@ucundinamarca.edu.co

**Abstract.** Fake news has become a social problem, exacerbated during the COVID-19 health crisis to the present. In this sense, the purpose of the article is framed in exposing how disinformation presents complex social topics, not only for the fact of spreading false or distorted information, but also for promoting hate speech that leads to acts of violence and social panic, questioning the decisions, political and health actions that have compromised social well-being. Under this view, fake news and video montages are characterized, which show to have a great impact on the credibility of scientific information, exacerbating problems whose dimensions in social and geopolitical terms demand prompt attention under regulations of a legal, technological, educational, and cultural.

**Keywords.** Artificial intelligence, COVID-19, deepfake, fake news, misinformation, pandemic, post-pandemic.

## 1 Introduction

The pandemic caused by COVID-19 unleashed a media frenzy that helped establish the alert on the part of the health sector to society, about prevention and self-care to minimize the risk of contagion.

However, part of this frenzy that began to circulate on the internet has not ceased to this day (overshadowed by the war between Ukraine and Russia, and the global economic crisis in force at the time of writing this article), through social networks , blogs, videos, newscasts and various information systems, which distorted the seriousness of the coronavirus and its variants, with content that attributed this disease as an urban myth, hoax, conspiracy orchestrated by governments of the great powers, world purge, a media to contract cancer or HIV, even as asymmetric warfare.

Because of the foregoing, it did not take long for rumors and hoaxes to surface through hashtags on social networks, blogs, memes, email messages, spam, video montages and photomontages of various kinds, catapulting misinformation through false news to such a level. that scientific information was compromised.

For example, associating the coronavirus with a simple flu, thus facilitating the contagion and rapid spread of the disease, with dire consequences of blurring the credibility of medical and scientific personnel [1]. Other facts related to fake news were the dissemination of erroneous medical information in the news and social networks [2], underestimating the impact of dissemination and care against COVID-19, added to publications asserting that young people and certain communities were immune under racist or xenophobic overtones, or that tropical countries should not worry because the problem was concentrated in those with seasons.

The most notorious case of spreading false news was carried out by the then President of the United States Donald Trump [3], who stated that the combination of hydroxychloroquine and azithromycin combated the coronavirus; the result was that many people believed it and consumed it, committing their lives in the process. In addition to this, the continuous messages of denial issued by this president through his Twitter account, minimizing the importance of taking prompt sanitary actions, unleashed such disinformation in the United States whose consequences were felt by all its inhabitants [4].

This behavior was joined by other political leaders in other countries, dismissing the pandemic to the point of not implementing biosafety and self-care measures, even not abiding by the restrictive measures necessary to stop the spread of the virus under questionable pseudoscientific arguments.

In accordance with the above, it is evident that disinformation has become a true virus, an infodemic, becoming the common denominator of the mass media, whose objective has been focused on generating sensationalism, mockery, disbelief and panic in the community, even inciting violence, hatred, and civil contempt. Added to this panorama is fraud, cyberpiracy, phishing and cyberattacks of various kinds that increased alarmingly during the pandemic and have become more pronounced in the post-pandemic.

## 2 Disinformation

The European Commission defines disinformation as "any form of false, inaccurate or misleading information, presented and promoted to cause public harm or for profit" [5]. Expanding this definition, disinformation is understood as that premeditated and sensational action that seeks to create instability in public opinion through disinformation rhetorical artifices such as the demand for disapproval, the appeal to authority and fear, stereotypes, and euphemisms, to the manipulation of emotions, to intentional imprecision, redefinition and revisionism that lead in many cases to extremism and identity theft.

Although misinformation is not new, the way it is used for its dissemination is, where Information and Communication Technologies (ICT) play a fundamental role in this regard. Digital resources have opened countless possibilities and opportunities for the transmission of information, for which freedom of expression in some contexts is distorted or misrepresented through stratagems without the common citizen often realizing it, making them believe lies or truths. half.

The problem lies in the fact that lies can lead to personal and/or popular beliefs, quickly establishing themselves in their social circles, thereby exponentially increasing misinformation, where the manipulation of public opinion affects both the image of people, institutions, or governments.

Terms such as "astroturfing, egging, trolls, catfish, click baiting or fake news were included in the warning lists of the media and social networking companies, drawing attention to the new digital-native misinformation" [6]. In the trajectory of the pandemic and post-pandemic, disinformation has been used as a method to create fear, chaos, and geopolitical and social instability, impregnated in many cases with virulent nationalism and populism, as is the case of authoritarian regimes such as China, Russia, and North Korea, among others, which manipulate social networks by controlling the news and messages of their citizens [7, 8].

Both misinformation and fake news have contributed to the emergence of cases of post-traumatic stress disorder [9]. The consequences of these disorders have not yet been measured in depth; however, significant repercussions are expected in the immediate future, because disinformation campaigns are not going to decrease, especially when their influence on public opinion is increasingly deep-rooted, added to the uncertainty about the control and definitive disappearance of the coronavirus in the medium and long term.

The problem of disinformation can reach such a harmful level causing irreparable damage to the real information, delegitimizing, or discrediting it, as well as those who support it. As a particular case, part of the proliferation of the coronavirus is attributed to disinformation, manifested through false news [10] combined with video montages, whose realism has made people believe that the information is legitimate, protected by rumors, decontextualization or pseudoscience, thereby promoting non-compliance with biosafety protocols, self-care, even refusing to get vaccinated.

These actions present different nuances about the behavior of people or communities in the face of a certain type of event, ranging from a simple joke to demonstrations against public, educational, sociocultural and health policies [11] to name a few. In addition, these actions can be carried out automatically using virtual robots (Bots) or by human activity (Trolls), whose purposes are more aimed at disseminating false information on a

massive scale, because behind it there is a lucrative business largely sponsored. part by the political and industrial sector [12]. The tendency to use bots as disseminators of false information lies in their autonomy, interacting with thousands of computer systems retweeting messages or publications through the hashtag. In the case of trolls, apart from being created and published by human hands, they use bots to amplify their dissemination.

In general terms, disinformation is tainted by false, misleading, and deliberate informative rhetoric artifices mediated by a sensationalist and media component. The actors responsible for these artifices are motivated by different causes, where the veracity of the news is not part of the minimum journalistic and informative values and standards, whose connotations in many cases present complex sociocultural and political nuances marked in most cases by psychosocial factors.

### 2.1 Fake News

Fake news is considered sensationalist or bullying information content, the dissemination of which is carried out through various media such as social networks, television, radio, portals, blogs, and the written press, among others. Fake news is characterized by lacking serious scientific and journalistic support; therefore, they are considered pseudo-journalistic content.

This form of popular expression is not new, it has simply been accentuated thanks to the different media whose creation, as [13] points out, "follows passion, politics, dissemination and payment". To these four P's are added panic, populism, power, parody, partisanship, provocation, and pseudo-journalism, which are complementary elements to this complex scenario forming a dynamic interaction as summarized in the following figure:

Fake news seeks to influence the community under the premise of misinformation, spreading under different nuances leading to generating social controversy, multiplying public opinion, stirring up spirits and satirizing. Likewise, they can contain a strong ideological component, often loaded with hate sponsored by third parties whose interest is to generate social destabilization.

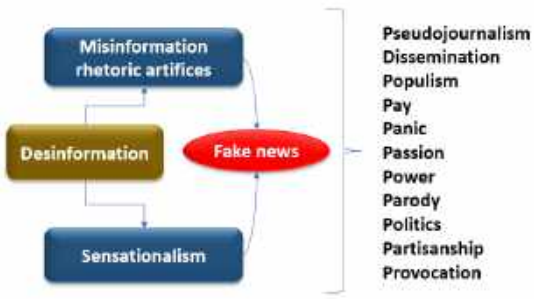



**Fig. 1.** Elements that make up misinformation and false news. Fountain. Own elaboration

In the same way, false news seeks to spread rapidly through ICT in a relatively short time, causing great impact depending on the context and content. Thus, the question arises, why are fake news so influential? The reasons are diverse and complex in many cases, due in part to the fact that false news can be created simply with the intention of having fun impregnated with black humor in some cases, which seeks to distort the real news or build the information based on assumptions and/or rumors from unreliable sources, such is the case of blogs and social networks.

Also, this news is based on a construct planned by an individual or group, tending to discredit, delegitimize and/or ridicule a person, community, industry, institution, or government; The reasons are diverse, as [14] points out: "satire or parody, misleading content, impostor content, fabricated content, false connection, false context and manipulated content".

In the case of the pandemic, the false news was initially directed at the medical community and scientific personnel, issuing confusing and inaccurate information, and in some cases contradictory information, fed by the media that, in search of the scoop and sensationalism, published news. decontextualized based on interviews whose information ranged from alarming to optimistic.

In addition to this, the information from non-specialist personnel increased misinformation and, incidentally, fed false news, showing nuances of humor, discontent, and skepticism.

In this order of ideas, the health sector went to the point of dismissing the severity of the pandemic, interpreting it as a seasonal flu [15] or a

disease typical of Asia. In this context, satires and parodies alluding to the coronavirus began, rapidly escalating to racial, misogynistic, xenophobic, and discriminatory stigma.

Parallel to this, the newscasts were not far behind, publishing that the virus was of artificial origin, weaving a misinformation ruse about conspiracies of all kinds and denial orchestrated by the WHO, Bill Gates and his Foundation, the World Bank, the Chinese government, universities, and research laboratories among others.

Otherwise, public figures and politicians fanned the flame of disinformation, neglecting sanitary measures, calling them unnecessary and ridiculous. The truth of all this was the seriousness and impact that the dissemination of false information brought with it, with the potential irreparable social and health damage in many cases.

### 2.2 Deepfake

The Deepfake (fake + deep learning, falsification using deep learning), has become a social-technological trend used to spread false information through video montages tricked with artificial intelligence (AI). A deepfake seeks to ridicule, defame, or distort real news regarding public figures, such as actors or politicians and to a lesser degree ordinary people, creating pornographic videos, satirizing campaigns, or interviews, among other actions, manipulating images and/or audio.

A particularity of a deepfake is its degree of realism that makes it more difficult to differentiate from a real video; this thanks to artificial intelligence that uses machine learning and deep learning algorithms under different development platforms such as generative adversarial networks (GAN) [16].

With GANs it is possible to obtain completely fictitious images, which allow launching disinformation campaigns and falsifying the profiles of certain specific targets on social networks in a short time. An example of a system that uses GAN is StyleCLIP [17, 18], which supports text-based image manipulation intuitively.

There are the programs that exchange faces or deep video portraits, which are the most common to apply deepfake, such is the case of the mobile applications FaceJoy, FacePlay and Reface; The first allows you to place a person's face in a fixed image, the second uses video templates to insert the face in them and modify them to see how it would look in other nationalities, and the third does the same as the previous ones, with the difference that You can insert a video in different contexts creating high quality montages.

Also, there is FaceMagic that allows you to create videos, photos, and gifs by modifying the face(s) instantly in different situations. Now, if you want to go much further by violating people's privacy, there is the DeepNudes web application, which with any image of a dressed woman – the suit doesn't matter – the system recreates her by removing the clothes, leaving her completely naked. The result is so realistic that it is difficult to differentiate that it is a false image [19].

The deepfake uses datasets, with images and audio related to the victim. The central idea is to train a neural network that mimics facial expressions, gestures, voice, and inflections with a high degree of accuracy, which are then simulated by inputting fake content. The problem with this technological development is that it can eventually be used for extortion, scams, and defamation. Faced with this fact, the authorities are increasingly concerned about the manipulation of information that can be used by anyone with minimal knowledge of video editing and specialized software management.

With the technology available and in progress, video montages will increase in the coming years, because not only ordinary people will be able to do it with minimal effort, but also groups that seek to generate political and social instability. Likewise, the generation of critical disinformation environments is expected, where political groups will use this type of resource to discredit their competition.

This leads to refining new techniques that not only improve lip synchronization, but also create facial maps with the help of AI of such quality that it will be very difficult to establish whether a video is fake. This scenario has made it a particular case that the US, through the Center for a New American Security (CNAS) organization [20], has repeatedly stated that the deepfake will become a potential threat to national security.

They could even interfere in future elections, an aspect that can be replicated for any country. The CNAS statement should not be taken lightly since the impact and damage that the deepfake is causing could even reach the point of creating disinformation wars.

The truth of all this is that it will demand new technological developments in the coming years that will allow it to go hand in hand with the technologies used in the deepfake. In this way, it entails a potential danger in a society that has been marked by hatred and resentment, since, by creating false information with political, religious, racist, conspiratorial epithets, etc., they can trigger conflicts of various kinds, very difficult. to control and deny.

A few years ago, the technology used to create the deepfake presented problems related to guaranteeing a credible reality in certain contexts, for example, a person's face or unnatural blinking at certain times. With artificial intelligence called DALL-E 2, DALL-E Mini, Gato, among others, these problems have already been solved, since these systems can generate photorealistic faces. These systems already have access to the public with certain restrictions, which in the future will most likely allow the faces of celebrities or public figures to be uploaded and manipulate the media through them.

The deepfake is a resource that can polarize information in a way equivalent to fake news, with the mitigation that video uses as the main resource, this being more persuasive during the era of disinformation.

The consequence of all this is that in times of pandemic and post-pandemic ICTs and emerging technologies such as artificial intelligence showed their negative side, unleashing waves of misinformation whose social impact has been significant to date, calling into question decisions and actions of the health, scientific and government authorities in matters of health, prevention, and vaccination.

Thus, both false news and the deepfake have become key resources for promoting disinformation on a large scale, producing uncertainty in society about what is credible and what is not. Its repercussions are notorious in various fields, being literally used as a cybernetic weapon of disinformation by certain communities and countries, which invest large amounts of technical, technological, and economic resources to manipulate public opinion.

### 2.3 Consequences of Misinformation

Disinformation in the health environment has been an aggravating factor since the start of the pandemic, continuing even in the post-pandemic period, manifested in different ways such as: propaganda, malicious intentional influence, conspiracy, stigma, politics, rumors, jokes, and even cyberattacks. Each of these forms act in a more direct and harmful way than another, in such a way that their social impact is relative in terms of diffusion and permanence.

For example, cybercriminals used disinformation [21] to create panic and fear during the COVID-19 outbreak, with the aim of launching social engineering, phishing, and malware distribution attacks to deceive and steal personal information and information. of hospital institutions and research centers. Also, these same strategies have been used by ordinary people to attack certain companies to hijack information through ransomware.

Due to the media impact that misinformation brings, it is difficult for it to disappear so easily; This is partly due to the human being's own culture of disclosing information, even if it is false, thereby expressing their feelings and disagreement with society.

Likewise, the media together with other ICT resources have been fundamental tools in terms of information concerning the health sector. However, it is these same means that have been used against them, overwhelming and confusing society with false or erroneous information, influencing public and political figures to such a degree that they have contributed to people's opinion being biased.

The good faith of those who trust information that is supposed to be from a reliable source is literally abused, the consequences of which have been to underestimate the risks of the coronavirus in terms of contagion [22], dissemination, and death.

Likewise, under the misinformation, thousands of people were encouraged to abuse the consumption of vitamins and the intake of toxic

liquids such as methanol, chlorine dioxide and sodium chlorite, among others, which supposedly treated the coronavirus.

Information was even published stating that the use of masks was correlated with cancer and that they were of no use in stopping contagion and, cases where the consumption of cow dung and other animals was recommended under the belief that it would cure them or would make immune.

Based on the horde of false news and disinformation spread everywhere, the authorities in the world chose to implement measures to refute them through official channels. However, the results have been unsuccessful so far due to the speed with which misinformation spreads, thus making its treatment complex.

In fact, it is possible that in the coming years there will be an increase in disinformation with geopolitical and conspiratorial nuances, marked by accusations against China and the WHO of the way in which the coronavirus emerged and its treatment in times of pandemic and post-pandemic, added to other sociocultural problems such as the increase in discrimination and xenophobia, among others.

From the foregoing it can be deduced that coordinated measures are required between the health sector, the media, governments, and international institutions, which monitor the information that circulates through different channels, guaranteeing its veracity and in the process curbing disinformation through three strategies, as shown in Figure 2:

1 Education: society needs to be educated in the management of real information, teaching it how to discard that which is not real. For this, online sources are available to confirm the information and educational technologies that can be quickly developed and implemented in various environments. In the same way, consider the ethics of information (fact-checking) leading to evaluating the overexposure of data online and discerning its value.

2 Technology: using various technological resources such as AI and the blockchain [23], they allow the veracity of a news item to be verified under certain parameters, which if not exceeded are deleted from the network. Also, the use of certificates such as AMP (Authentication of Media via Provenance) is proposed, which validate the information both from the device with which the image was captured or recorded a video, to the editing tools used by professionals and companies.

3 Regulations: social networks are one of the means of communication on which the greatest responsibility falls for acting as a means of disseminating false news, therefore, the existence of a regulatory control that restricts this procedure without going against the right to free expression is necessary.

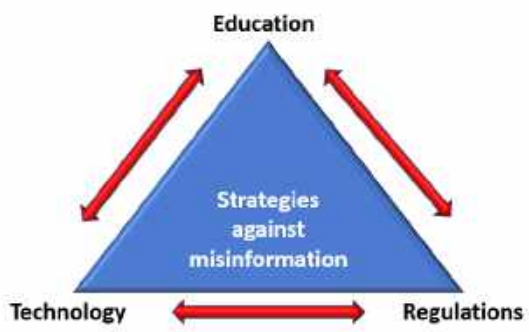



**Fig. 2.** Strategies against misinformation. (Own elaboration)

These strategies are viable if they are coordinated and implemented in a timely manner using the technology at hand. In this regard, there are official companies whose job it is to create synthetic data that allow training deep learning models leading to avoiding difficulties in obtaining real data on humans.

Part of this type of development presents some variants that are the use of antagonistic generative networks (GAN); which allow the creation of high-quality fake images and videos, where superimposing the faces of celebrities or politicians on porn stars is relatively easy [24], including superimposing biometric data. This panorama leads one to think that in a relatively short time being able to discern what is real or false will be in the hands of AI, which is a major social and judicial problem, among other factors.

Faced with the above, the use of C2PA certificates (Coalition for the Origin and Authenticity of Content) [25] is proposed as an industry standard, whose owner is Adobe,

guaranteeing consumers when they consult a website that the Graphic information such as photos and videos is true, although the download cannot be avoided to be manipulated, at least traceability of its original source can be carried out and a deepfake can be denied.

It should be clarified about the C2PA, that this "does not say anything about the veracity of what is represented in an image, it indicates when something was done and who did it: when it was produced, how it was edited, published and reached the consumer" [26], all through a certified watermark of the metadata. What is expected in the short term is that the manufacturers of cameras and smartphones adopt C2PA, then gradually escalate the staff that create the content and the chain of services so that they do so as well, as is the case of social networks.

Other alternatives related to C2PA is Project Origin [27], which is an alliance of companies such as the BBC, the New York Times, the CBC/Radio-Canada, Microsoft and Spotify, among others, that use a development of this last company called Authentication of means of origin; This seeks to verify the publications of images, videos and audio, through the analysis of metadata and cryptographic hashes of each file, which when manipulated automatically changes its digital footprint. These alternatives are not available to the public since information would be given to the creators of fake news tools to counteract them.

### 2.4 Infodemic and Health Implications

Fake news in times of pandemic and later in vaccination campaigns, caused an exponential growth of misinformation, a phenomenon that is summarized in the term infodemic [28]. The overwhelming nature of this situation, despite the number of infected and deaths, the infodemic has affected the credibility and trust of society in the face of health institutions and their members, as well as the prevention and vaccination policies proffered by governments.

The degree of affectation has been such that the information circulating on the internet questions the effectiveness of the vaccines, encouraging people not to get vaccinated under the argument that it is a massive purge, or that its objective is to sterilize the population and thereby stop the explosion. demographic among other nonsense statements, in addition to the instigation not to wear a mask, to consume unapproved alternative medications, alcoholic beverages and concoctions, among hundreds of other hoaxes.

Another topic of the infodemic is related to publications, where authors have used the relationship between COVID-19 and the immune system as a hook to promote articles or books whose scientific support is debatable, such has been the case of ancestral medicine texts, which hypothetically prevent and they treat the coronavirus, making people believe that they are the salvation table for their ills. These types of actions are considered elaborate fake news, which in many cases is supported by rulers and even by unqualified medical personnel.

It is difficult for the citizen in general to decant the rain of misinformation that they constantly receive and that can eventually become dangerous, thus dismissing the scientific studies and medical care that has led an infected individual to focus their efforts on herbs and home remedies. exposing not only your life, but those around you, becoming a potential spread vector for your community. These actions are possibly reprehensible; however, the direct blame falls on those who publish false information without measuring the consequences of their actions, taking advantage of the chaos, the despair of people and society in general.

With the technological resources available to society, the proliferation of the infodemic has worsened, this due to the increase in internet connectivity and access to various free or low-cost digital resources, which allow the editing and manipulation of images, texts, and audios with total freedom. Consequently, the publication and dissemination of false information has various channels, most of which have proven to be impossible to control forcefully.

As a particular case, social networks have proven to be extremely difficult to regulate, which have become a focus of violence and digital intolerance, despite unsuccessful attempts to prohibit and eliminate such actions. The problem is further exacerbated when influencers manipulate their audience in an irresponsible way without considering the social damage that their actions cause. Additionally, with AI and machine learning

applied to obtain geospatial data, huge amounts of information can be extracted, which a few years ago access was exclusive to intelligence agencies, thus allowing them to be manipulated at their convenience. who have the technical and technological resources for this purpose.

## 3 Discussion

Can you trust what you are seeing through the different media? The answer to this question is complex, false news disseminated through the media has repeatedly become a social evil, causing, apart from misinformation, problems of a diverse sociocultural and geopolitical nature, significantly affecting the health sector. Worldwide. For example, a false news item that had a high social impact was that of then-President Donald Trump, who stated that by ingesting bleach he could fight the coronavirus in a very short time.

The result of this news was that several people were hospitalized for ingesting detergent. This publication generated strong controversy and rejection from the medical community, which was later corrected not by Trump but by the New York Department of Health. Similarly, leaders of other countries underestimated contagion and self-care, whose example was seconded by the population with consequences already known by public opinion.

The foregoing highlights the effects of politics in the era of misinformation, which has not gone unnoticed regarding the issue of vaccination, questioning its health function, arguing that the time required for testing and obtaining the vaccine per it is doubtful, or that they are genetic markers equivalent to those used in cattle, among other unsubstantiated assertions.

The tragedy of this situation is the presence of personal, economic, and geopolitical interests. In addition to this, there are organized groups on the Internet whose purpose is to promote violent and intolerant hate messages, which are difficult to eradicate from the web due to their great recovery capacity [29].

Misinformation and false news have dire consequences for the credibility of society, added to the probability that it will worsen in the coming years; this is partly due to the technological evolution itself that facilitates the proliferation of digital tools such as portals that allow the creation of false news, or AI-based systems using algorithms and bots, which ironically can be used to counter disinformation by filtering it.

In the case of AI, there are advanced platforms available to the public such as GPT-3 and GPT-4 (Generative Pre-trained Transformer), which theoretically can indirectly create fake news (manipulating it with counterintuitive rhetorical arguments) and cyber-hooks or clickbaits [30] of such a degree of quality that it will be difficult to know if they are real or not.

Similarly, there is the BLOOM system (BigScience Large Open-science Open-access Multilingual Language Model) [31], much more advanced than GPT-3, capable of generating text in 59 languages, and the recent ChatGPT based on GPT- 3. These tools present unprecedented potential in technological terms, raising the debate about their social implications if they are manipulated to create false information.

On top of GPT-4, made up of approximately 100 trillion parameters as opposed to its GPT-3 counterpart of 175 billion parameters, it makes for a robust system that allows you to run generative algorithms with the ability to create realistic information and videos of virtually anything. In the case of fake news, it is only a matter of time before false information about anyone can be created with few images.

This leads to elucidate that, with the continuous advancement of AI and the creation of a public database of faces, it will be possible to create the image of a fictitious person and from there assemble a whole paraphernalia on it with the aim of using it in acts. illicit Although it seems science fiction, the possibility is given, and the worst of all this is that it will be very difficult to differentiate if this "avatar" is human or not.

It can be affirmed that before disseminating unreliable information, it is essential to previously confirm its veracity, comparing it with credible sources. In the same way, assume a critical role when consulting information and questioning what is read, for this, it is suggested to consult web portals that help to deny false news such as: WHO, Verified (ONU), European Center for Prevention and Control of diseases, 1de2.edu, Truly media, FactCheck.org, Botometer, Snopes, Hoaxy,

Fighting disinformation, Credibility Coalition (CredCo), Google News Initiative, Knight Foundation, W3C Credible Web Community Group, etc. As a particular case, the WHO has established campaigns aimed at identifying and reporting incorrect or misleading information that is potentially harmful in times of pandemic and post- pandemic.

Regarding the role of the media, it is necessary to be focused on maintaining the credibility, honesty, transparency, and trust of the news disclosed to the community, where the work of providing verifiable information to those who need it must be present. Information verification or fact-checker [32] does not conflict with freedom of the press, in fact, there are online resources that allow this task to be carried out quickly and easily.

However, although these resources partially counter disinformation and false news, it is necessary to formulate regulations that allow a brake on its different modalities and contexts, in a similar way to what was done by Singapore [33], but without going to the ends.

There are initiatives in this direction, but there is still a lack of technical and technological tools to facilitate this work, such as content verification systems, designed under international standards that provide security without violating freedom of expression. This task falls not only on the legislative and technological entity, but on the community in general, where respect for others prevails without violating fundamental rights.

As additional information, to verify the validity of a piece of news and counter misinformation, it is recommended:

1 Extrapolate from the news prejudiced aspects marked by hate speech, racist, misogynist, sexist, xenophobic and discriminatory.

2 State entities must appropriate the problem by removing from the web those networks or groups that promote hate and misinformation. Similarly, create specialized groups that counteract the activities of those who promote misinformation and intolerance.

3 Penalize and prosecute those who create and spread false news. It is important to note in this regard that there are criminal laws, whether for those who publicly encourage, promote, or directly or indirectly incite hatred, hostility, discrimination, violence, and crimes of genocide against humanity, among others, through any technological means. The reasons are the order of the day, since these range from racism, anti-Semitism, through personal or group ideology, religion, or beliefs, to situations of a family, ethnic, race or national nature.

4 Combine efforts between government entities, researchers, and technology companies, leading to developing coordinated technical and technological strategies that allow counteracting disinformation and all kinds of actions that violate human rights.

5 Promote research leading to disseminate science more assertively to the public. [34] cites Caulfied (2020), who states that "researchers need to have a greater presence on social networks and disseminate science in the same symbolic battlefields that are currently occupied by disinformation".

6 For society in general, try to be as objective as possible without intervening emotions. This leads to thinking twice before forwarding information to third parties without having verified its source.

7 Compare the information with reliable sources: indexed articles, renowned newspapers, and news, antifake news web portals, etc.

8 Look at the publication date of the news, use online image and video verifiers, because it often happens that information is recycled to support false news.

9 Collaboration of various organizations such as the WHO, Google, Tencent, Twitter, Meta, Pinterest, TikTok and YouTube, among others, to focus efforts on denying hoaxes and demystifying false information.

10 Transparent dissemination of reliable scientific data by governments, publishing them proactively under laws and policies concerning the right to information.

11 Expand the Services provided by platforms such as the UN's "Verified" (https://shareverified.com/es/), whose objective is focused on providing reliable and accurate information about the coronavirus among other aspects of social interest.

12 Promote initiatives such as Eleuther AI [35] characterized by its research on the interpretation, security, and ethics of large AI-based language models, which can be used for criminal purposes such as the creation and dissemination of false news or deepfake.

13 There are technical solutions using AI through machine learning [36], deep learning and natural language processing using methods such as logistic regression, the Naive-Bayes multinomial model and support vector classifiers (SVM) pretty good at dealing with the topic at hand. For example, there are already automated detection systems that filter false information and reinforce what is real. In this same direction, AI is used to eliminate bots that can be used to carry out distributed denial of service (DDoS) attacks [37].

As a complement to the above, Google has become a tool that allows you to verify if an image is false or not. The procedure is simple, just look at a fact verification label that appears at the bottom of the image results. When selecting this label, a summary is displayed that shows the authenticity of the image through the verification of the web page where it is published.

Intel has also created the AI-based Deepfake Detection System Fake Carcher that detects in real time if a video is fake, with an accuracy rate close to 100%, through the analysis of subtle variations in the color of the pixels. related to blood flow signals from a person's face.

As can be seen, the role of AI in disinformation is a fact, it only remains to polish certain algorithms so that automatic systems are programmed to such a level that it is difficult to distinguish between real and false information. In the case of fake news, thanks to advances in AI, particularly algorithms related to natural language (NLP) [38] such as the one used in the GPT-3 and 4 system [39], could be used for this purpose with a high degree of credibility, even supported by reliable sources created or decontextualized.

GPT-4 is not the only most powerful AI tool at the time of writing, as there is another similar AI, called Google Switch Transformer with 1.6 billion parameters and WuDao 2.0 (China), which is 10 times more complex than the GPT-3. Another type of AI that has aroused interest due to its release of code and free access is OPT (Open Pretrained Transformer) from the Meta company, which theoretically will only be used exclusively for research, however, questions are raised about its potential generation of false information. and racist language, just like LaMDA; an AI model for dialog applications.

Thus, AI can have a great social impact, such is the case of geopolitics, where the manipulation of information and censorship can undermine credibility, for example, the war between Ukraine and Russia [40], or the pandemic spread of Covid variants in China. It is no longer a secret that AI allows promoting false news, and at the same time, there are other AIs that allow evaluating the veracity of the information, examples of this are: Fake News Challenge, Spike, Snopes, Hoaxy and Deep Entity Classification (DEC), among other.

The world faces an uncertain future scenario in terms of massive false information, which leads us to think that society will be subject to perceiving information in a fractional way, with the risk of assuming radical positions that possibly lead it to make bad decisions, under polarized versions and strategic propaganda aimed at destabilizing the sociocultural and geopolitical construct, among others. The challenge of fighting against disinformation is ongoing, demanding that measures be taken to guarantee that freedom of expression is not misrepresented and leads society to greater disinformation chaos with the corresponding consequences.

Similarly, in health matters about demystifying what circulates through the networks and other channels about vaccines against Covid-19, [41] points out "that the authorities must be very clear about what is known and what is not it is known, involve the public in the debates, take their opinions seriously and build trust through transparency", this action will undoubtedly contribute to reducing uncertainty in society.

Added to the above is the massive manipulation of information, which has proven to be very dangerous because it moves large masses of people. In a localized way, fake news is being used as an attack on corporate targets, where a meeting can be silently recorded and the information then manipulated to incorporate false and/or damning data that, if leaked, would create commercial

shock, thereby compelling to the company paying for not publishing.

Disinformation is not only concentrated in the written word and video, but also in sound, such is the case of an AI belonging to the company Sonatic [42], whose development uses the voices of artists, this focused on the field of entertainment. However, this same technology can be used to generate voices for deceptive purposes.

A particularity of this AI is that it can be configured so that the voice adjusts to a tone of fear, sadness, anger, happiness, and joy, added to the fact that it can express ironic tones, shyness and even boast. As if that were not enough, the voice can be modulated in such a way that certain voice patterns such as tone, intensity and vocalization can be modified, giving the listener the sensation of a smiling tone, accompanied by breaths in an equivalent way to what he does a human being.

## 4 Conclusions

Disinformation in the health sector in times of pandemic and post-pandemic created all kinds of social and geopolitical tensions that have demanded regulatory, technical, and technological actions to mitigate it. The problem may become more acute in the coming years, due in part to the continuous advances in AI that will make it more difficult to unmask false news or false videos, added to multisectoral cyberattacks promoted by criminal gangs and the business that revolves around the dissemination of false information. deliberate misinformation.

Fake news will not end unless there are policies that punish those who promote misinformation. However, fake news is another matter, as it is a disruptive movement that brings with it profound implications in the various tabloid and corporate media, based on the technology used to create it. This brings consequently, that the journalistic and judicial evidence with which it is currently judged based on audio and video will be refuted, because they cannot be undeniable evidence of a fact.

As if this were not enough, in this context, AI is going to assume a technological role that debilitates justice. However, alternatives are presented that allow validating the authenticity of audio and video files such as the use of the blockchain.

The problem is in the technological implementation with the corresponding costs, plus the deception technologies advance much faster than those to counteract them.

Disinformation and the flow of false news issued by the different communication media must be considered as a problem and risk of high social, geopolitical, health and security impact, and as such the legislation must act accordingly to minimize them.

The reason lies in the fact that the information that society receives is often out of context and/or distorted, whose reflection on its veracity and reliability is debatable. This scenario has the effect that many people are susceptible to being influenced by misinformation, since in one way or another they confirm their beliefs, which are reinforced by becoming recurrent in different media until they become convincing, completely distorting the truth.

Fake news spreads much faster than the real ones [43], therefore, it is advisable to be aware of those that arrive by different means and before forwarding them to third parties, consider the implications in terms of violating the honor, image and good name of a person, community, or institution.

## References

1. **Cifuentes, F. J. (2020).** Fake news during COVID-19: how to detect them? Comunicación, No. 42, pp. 100–103. DOI: 10.18566/comunica.n42.a07.
2. **Palomino, G. M., Lovón, M., Arellanos, R. (2020).** La red sanitaria y su participación en la difusión o contención de las fake news y bulos relacionados con la COVID-19: el caso de Lima-Perú. Revista Latinoamericana de Comunicación, No. 145, pp. 89–106.
3. **Rosenberg, H.; Syed, S., Rezaie, S. (2020).** The twitter pandemic: The critical role of twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic. Canadian Journal of Emergency Medicine, Vol. 22, No. 4, pp. 418–421. DOI: 10.1017/cem.2020.361.

4. **Pérez, D. J., Meso, A. K., Mendiguren, G. T. (2020).** Fake news y coronavirus: detección de los principales actores y tendencias a través del análisis de las conversaciones en twitter. El profesional de la información, Vol. 29, No. 3, p. e290308. DOI: 10.3145/epi.2020.may.08.
5. **Dias-da-Silva, M., Walmsley, A. (2019).** Fake news and dental education. British Dental Journal, Vol. 226, No. 6, pp. 397–399. DOI: 10.1038/s41415-019-0079-z.
6. **Romero, R., Rodríguez, H. (2019).** Desinformación y posverdad en los medios digitales: del astroturfing al click-baiting. Romero, R. L. y Rivera, R. D. (Coord.), La comunicación en el escenario digital Actualidad, retos y prospectivas, Ed. Pearson Educación, pp. 379–401.
7. **Diresta, R., Miller, C., Molter, V., Pomfret, J., Tiffert, G. (2020).** Telling China's story: The chinese communist party's campaign to shape global narratives. Hoover Institution, Stanford Internet Observatory.
8. **Horowitz, C., Allen, C., Saravalle, E., Cho, A., Federick, K., Scharre, P. (2018).** Artificial intelligence and international security. Washington, DC. EE.UU., CNAS, CNASDC.
9. **Dong, M., Zheng, J. (2020).** Letter to the editor: Headline stress disorder caused by netnews during the outbreak of COVID-19. Health Expectations: An International Journal of Public Participation in Health Care & Health Policy, Vol. 23, No. 2, pp. 259–260. DOI: 10.1111/hex.13055.
10. **Ramón, F. F. (2020).** Comunicación y noticias falsas en relación al COVID-19: algunas reflexiones sobre la información, la desinformación y propuestas de mejora. Revista española de comunicación en salud, Suplemento, Vol. 1, pp. S253–S264. DOI: 10.20318/recs.2020.5375.
11. **Gatica-Mancilla, S. E. (2021).** Fake news, trolls y otros encantos. Cómo funcionan (para bien y para mal) las redes sociales. Comunicación y medios, Vol. 30, No. 43, pp. 168–169. DOI: 10.5354/0719-1529.2021.61086.
12. **Muñoz. M. C. (2020).** Noticias falsas: Confianza y configuración de la opinión pública en los tiempos de internet. El Cronista del Estado Social y Democrático de Derecho, No. 86-87, pp. 122–138.
13. **Wartdle, C. (2017).** Noticias falsas. Es complicado. First Draft, Harvard Kennedy School's Shorenstein Center, https://es.firstdraftnews.org/2017/03/14/noticias-falsas-es-complicado.
14. **Tagliabue, F., Galassi, L., Mariani, P. (2020).** The "Pandemic" of disinformation in COVID-19. SN comprehensive clinical medicine, No. 1–3. Advance online publication. DOI: 10.1007/s42399-020-00439-1.
15. **Alqahtani, H., Kavakli-Thorne, M. Kumar, G. (2021).** Applications of generative adversarial networks (GANs): An updated review. Archives of Computational Methods in Engineering, Vol. 28, pp. 525–552. DOI: 10.1007/s11831-019-09388-y.
16. **Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D. (2021).** StyleCLIP: Text-driven manipulation of StyleGAN imagery. rXiv:2103.17249v1 [cs.CV].
17. **Singh, A. (2021).** Guide to StyleCLIP: Text driven image manipulation. https://analyticsindiamag.com/guide-to-styleclip-text-driven-image-manipulation/.
18. **Pérez, E. (2022).** DeepNude: la polémica aplicación que "desnuda" a cualquier mujer mediante inteligencia artificial. https://www.xataka.com/privacidad/deepnude-polemica-aplicacion-que-desnuda-a-cualquier-mujer-mediante-inteligencia-artificial.
19. **Horowitz, M., Allen, G., Saravalle, E., Cho, A., Frederick, K., Kania, E., Scharre, P. (2018).** Artificial intelligence and international security. Washington D.C. EE.UU. Bold Innovative, Bipartisan.
20. **Lalupa (2021).** Las 10 recomendaciones y predicciones sobre ciberseguridad más importantes para 2021. https://lalupaonline.com.ar/?p=168922.
21. **MaassenVanDenBrink, A., de-Vries, T., Danser, A. H. J. (2020).** Headache medication and the COVID-19 pandemic. Journal Headache Pain21, Vol. 38. DOI: 10.1186/s10194-020-01106-5.

22. **Huckle, S., White, M. (2017).** Fake news: A technological approach to proving the origins of content, using blockchains. Big Data, Vol. 5, No. 4, pp. 356–371. DOI: 10.1089/big.2017.0071.
23. **Hao, K. (2021).** Estos humanos falsos anuncian una nueva era para la IA. https://www.technologyreview.es/s/13449/estos-humanos-falsos-anuncian-una-nueva-era-para-la-ia.
24. **Ortiz, R. F. W. (2021).** El proyecto C2PA y sus implicancias con el derecho a la libertad de pensamiento y expresión. Derecho y Cambio Social. https://www.derechoycambiosocial.com/anexos/MISCELANEA/2021/El_proyecto_C2PA.pdf.
25. **Branscombe, M. (2021).** Deepfakes: Microsoft and others in big tech are working to bring authenticity to videos, photos. Deepfakes: Microsoft and others in big tech are working to bring authenticity to videos, photos – TechRepublice.
26. **England, P., Malvar, H., Horvitz, E., Stokes, J., Fournet, C., Burke, R., Chamayou, A., Clebsch, S., Costa, M., Deutscher, J., Erfani, S., Gaylor, M., Jenks, A., Kane, K., Redmiles, E., Shamis, A., Sharma, I., Wenker, S., Zaman, A. (2021).** AMP: Authentication of media via provenance. Proceedings of the 12th ACM Multimedia Systems Conference, pp. 108–121. DOI: 10.1145/3458305.3459599.
27. **Arroyo-Sánchez, A. S., Cabrejo-Paredes, J. E., Cruzado-Vallejos, M. P. (2020).** Infodemia, la otra pandemia durante la enfermedad por coronavirus. Anales de la Facultad de Medicina, Vol. 81, No. 2. DOI: 10.15381/anales.v81i2.17793.
28. **Derzsy, N. (2019).** Strategies for combating online hate. Nature, Vol. 573, pp. 203–204. DOI: 10.1038/d41586-019-02447-1.
29. **Bazaco, A., Redondo, M., Sánchez, G. (2019).** El clickbait como estrategia del periodismo viral: concepto y metodología. Revista Latina de Comunicación Social, Vol. 74, pp. 94–115. DOI: 10.4185/RLCS-2019-1323.
30. **Gibney, E. (2022).** Open-source language AI challenges big tech's models. Nature, Vol. 606, pp. 850–851. DOI: 10.1038/d41586-022-01705-z.
31. **Ufarte, R. M., Murcia, V. F. (2018).** El fact checking: en busca de un nuevo modelo de negocio sostenible para el periodismo. Estudio de caso de Miniver, en Miguel Hernández, Communication Journal, Vol. 9, No. 2, pp. 511–534. DOI: 10.21134/mhcj.v0i9.267.
32. **Zastrow, M. (2019).** Singapore passes 'fake news' law following researcher outcry. DOI: 10.1038/d41586-019-01542-7.
33. **Sánchez, T. N. (2020).** Desinformación en tiempos de COVID-19: ¿Qué podemos hacer para enfrentarla? Revista Cubana de Información en Ciencias de la Salud, Vol. 31, No. 2, pp. e1584.
34. **Smith, G. S. (2022).** EleutherAI: When openAI Isn't open enough this group's free and open-source AI language model aims for GPT-3 power with Linux-scale collaboration and distribution. https://spectrum.ieee.org/eleutherai-openai-not-open-enough.
35. **Ahmed, H., Traore, I., Saad, S. (2017).** Detection of online fake news using n-gram analysis and machine learning techniques. In: Traore I., Woungang I., Awad A. (eds) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments, ISDDC 2017, Lecture Notes in Computer Science, vol 10618. Springer, Cham, pp. 127–138.
36. **Márquez, D. J. (2020).** Internet of things and distributed denial of service as risk factors in information security [Online first]. IntechOpen, DOI: 10.5772/intechopen.94516.
37. **Bisk, Y., Zellers, R., Le-Bras, R., Gao, J., Choi, Y. (2019).** Piqa: Reasoning about physical commonsense in natural language. arXiv preprint arXiv:1911.11641.
38. **Floridi, L., Chiriatti, M. (2020).** GPT-3: Its Nature, Scope, limits, and consequences. Minds and Machines, Springer, pp. 1–12. DOI: 10.1007/s11023-020-09548-1.
39. **Corral, H. D. (2022).** Medios de comunicación en la guerra de Ucrania, voces y certeza frente al silencio y la desinformación. Documento de Opinión, IEEE. Vol. 47.

**40. Subbarao, K. (2020).** COVID-19 vaccines: time to talk about the uncertainties. Nature, Vol. 586, pp. 475. DOI: 10.1038/d41586-020-02944-8.

**41. Hernández, G. (2022).** Esta voz generada con inteligencia artificial suena como Scarlett Johansson y es capaz de coquetear y hasta burlarse de su interlocutor.

**42. Vosoughi, S., Roy, D., Aral, S. (2018).** The spread of true and false news online. Science, Vol. 359, No. 6380, pp. 1146–1151. DOI: 10.1126/science.aap9559.

Article received on 21/02/2023; accepted on 19/02/2024.
*Corresponding author is Jairo Eduardo Márquez-Díaz.

ISSN 2007-9737

# A Fuzzy Inference Model for Evaluating Data Transfer in LTE Mobile Networks via Crowdsourced Data

Julio Ernesto Zaldivar-Herrera[1], Luis Pastor Sánchez-Fernández[1,*], Luis Manuel Rodríguez-Méndez[2], María Teresa Zagaceta-Álvarez[2]

[1] Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico

[2] Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica, Mexico

{jzaldivarh2021, lsanchez}@cic.ipn.mx, {lrodriguezm, mzagaceta}@ipn.mx

**Abstract.** Improving the Quality of Service (QoS) in the data transfer of 4G Long-Term Evolution (LTE) mobile networks has been a significant concern. Previous analyses have focused on enhancing network infrastructure using statistical tools, computational algorithms, and fuzzy models to improve mobile network operators. Those works are based on simulated data or data collected by a specialised modem without providing user information. In this study, we propose a fuzzy inference model to evaluate QoS Key Performance Indicators and signal parameters using data acquired by user equipment through collaboration or crowdsourcing. This fuzzy inference model provides specialists with a new method for assessing the QoS and offers users relevant information on the quality of data transfer service in LTE networks. The evaluation is based on fuzzy QoS, and effectiveness indices are classified into five levels: Very poor, Poor, Acceptable, Good, and Very good. Furthermore, the model can evaluate other data samples different from those used in this proposal. Finally, this method can assess the data transfer of 5G networks, making respective adaptations.

**Keywords.** Quality of service, key performance indicators, long-term evolution, crowdsourcing, fuzzy inferences system, assessment indices.

## 1 Introduction

As technology continues to develop, the demand for activities and the number of mobile devices have increased, resulting in a significant increase in data transfer over mobile broadband networks [1, 2, 3].

Consequently, there has been a rise in the use of Over-the-Top services [4]: video, audio, voice, or data applications transmitted over fixed or mobile internet platforms [5]. User can now access various multimedia applications through their devices, requiring reliable connectivity at any time and place.

This translates into the need for adequate Quality of Service (QoS) from mobile network operators (MNOs) [6]. Technological advances in multimedia offerings have forced operators to adopt a user-centric and quality of experience (QoE) approach [7].

Telecommunications regulatory bodies recommend evaluating QoS. However, ensuring QoS in the 4G mobile network is a significant challenge due to constant changes in the network [8]. The International Telecommunication Union (ITU) mentions that the QoS planned by network operators is typically different from the level users experience and could even be much lower than expected [9].

QoS is a set of measurable quality parameters called key performance indicators (KPIs). These indicators provide the necessary information for planning, performance analysis, and network optimization and can be either technical or non-technical [10]. As mentioned by the ITU, examples of technical KPIs are call success rate, call drop

rate, and upload and download connection speeds, among others.

Non-technical KPIs are customer-focused and may include parameters such as billing accuracy and failures [9]. Inadequate KPI data can limit service efficiency, increase operating costs, and negatively affect users [10]. Correct data is beneficial to network operators and users, as well as for research purposes. For example, an ensemble learning scheme for indoor-outdoor classification based on cellular data from a commercial LTE mobile network has been presented in [11], where the data was obtained with a single-user equipment (UE) model.

Numerous studies have focused on LTE mobile networks' data transfer KPIs and QoS. A comprehensive guide to standardized QoS assessment models is presented in [12]. Graphical comparisons of KPIs with received signal parameters have also been performed [13].

However, this study does not consider obtaining an index to determine the QoS of the LTE mobile network. On the other hand, to increase revenues from network services provided by the MNOs, the base station (eNodeB) performance is evaluated when a higher priority QoS is enabled for some LTE users [14]. In that study, the commercial network and UE were used. The study mainly focused on improving the infrastructure of the MNOs, but it is a very relevant work for the 4G network in general.

Other studies focus on the development or use of LTE mobile network simulators. In those studies, the network capacity is calculated. Improvements in the scheduling process for the network are proposed to impact both the QoS and QoE, thus increasing the spectral efficiency in terms of network throughput or packet delay prediction [7], [15, 16, 17, 18].

In addition, other research works aim to improve QoS and QoE during video transmission over LTE networks by evaluating and analysing various configurations and parameters [19 20 21 22 23]. Although, in these studies, there is an interest in increasing QoS and QoE, their main focus remains improving the services of the MNOs.

Several studies use experimental data collected from a smart city to conduct statistical analysis on the QoS of mobile networks. They present statistical descriptions and probability distribution functions for the KPIs to aid comprehension [10]. However, the experimental data contains some missing values estimated for each network parameter using the PCHIP algorithm and statistical error analysis [24]. The measurements in both cases were taken using a specialised modem. The Egil model [25] is also proposed to estimate signal loss using the quadratic regression method. While these investigations offer valuable statistical insights for the 4G LTE mobile network, they require a large data sample, which the current proposal aims to avoid.

The scientific literature encompasses various computational models, including a gradient-based iterative process to determine the optimal tilt configuration for the LTE eNodeB antenna [25].

Several sets of rules have been proposed to optimise resource allocation in downlink scheduling, and their performance has been evaluated by comparing the Knapsack and Priority-only algorithms [26]. The QoS-aware downlink scheduling algorithm (QuAS) was presented in [27] to enhance the QoE for peripheral users. Additionally, an innovative approach aims to maximise QoE by sharing an available channel among video traffic flows, incorporating genetic algorithms and random neural networks [23].

Despite these efforts offering alternatives to improve QoS and QoE, it is necessary to present relevant information to the user, such as knowledge of the actual QoS and effectiveness that users are experiencing.

On the other hand, Zadeh's fuzzy set theory is an extension of classical binary logic and has had a lasting impact on artificial intelligence [28]. The essential advantage of the fuzzy approach over binary logic lies in its flexible decision boundaries, providing greater adaptability to specific application domains [29].

Fuzzy logic draws inspiration from our understanding of human cognition in decision-making systems, making it widely accepted as an explainable artificial intelligence among interdisciplinary experts. It effectively deals with input variables, their ranges, limits, and variations, facilitating design, verification, and continuous improvement [30, 31, 32].

In the context of LTE networks, fuzzy systems have been employed in Call Admission Control to

reduce call drop and call blocking probabilities, as well as mitigate co-channel interference [23, 33] and [34].

To reduce costs and minimize negative impacts on the user experience of self-organizing networks, a problem-solving rule learning method based on fuzzy logic controllers and data mining techniques was proposed in [35].

To assess the QoS of LTE networks, particularly during the handover process, a Sugeno-type fuzzy model was employed to analyse four QoS KPIs across four applications [36]. Furthermore, a novel study compared the Sugeno-type fuzzy model against an Adaptive Neuro-Fuzzy Inference System (ANFIS) model tailored explicitly for QoS calculations in heterogeneous LTE networks [37].

However, it is worth noting that both investigations relied on simulated data and failed to propose a fuzzy index that effectively correlates KPIs with received signal parameters. As a result, users should receive more easily understandable and explainable information regarding the LTE network's QoS.

This paper introduces a novel fuzzy inference model for evaluating QoS and data transfer efficiency in LTE mobile networks. It incorporates two user-friendly fuzzy indices for swift categorisation of QoS and effectiveness, benefiting both network evaluators and end-users. The information repository used to evaluate the fuzzy model was collected through crowdsourcing.

The structure of the paper is as follows: Section 2 presents the methodology and information repository, followed by the results in Section 3, a comprehensive discussion in Section 4, and concluding remarks in Section 5.

## 2 Materials and Methodology

This section shows the phases that were attended to develop the fuzzy inference system. Fig. 1 introduces the methodology of our proposed model inspired by [38] and features six distinct stages: Fig. 1(a) acquisition of an information repository compiled through crowdsourcing; Fig. 1(b) selection of KPIs and signal parameters to be used as input variables, alongside data filtration to discard any outliers; Fig. 1(c) proposal of membership functions for the input variables and two fuzzy indicators; Fig. 1(d) creation of the fuzzy rules considering six input variables for QoS and two for effectiveness; Fig. 1(e) implementation of defuzzification using the Centroid Method; and Fig. 1(f) obtaining fuzzy indices for both QoS and effectiveness.

Further explanation of these stages can be found in subsections 2.1-2.5.

### 2.1 Information Repository

The information repository was gathered using crowdsourcing by teams of users equipped with mid-range mobile phones (UE) in the central Alameda zone of Mexico City during the first two months of 2021. This information was facilitated by the Telecommunications Engineering branch of the Postgraduate Sciences program (PCIT), part of the Postgraduate Studies and Research Section (SEPI) at the Higher School of Mechanical and Electrical Engineering (ESIME), Zacatenco unit of the National Polytechnic Institute (IPN).

Crowdsourced measurements from an end-user perspective prove crucial in enhancing the overall QoS, facilitating the acquisition of valuable information beyond the mere network layer and into the user and application layers. This approach allows a deeper understanding of any challenges or quality issues users face within the network [39].

Mid-range phones during this data-gathering process do not influence or impact the fuzzy indices obtained in this work. Device range classification is based primarily on RAM, screen resolution, and processor performance.

On the other hand, the transceiver and antenna are similar across all mobile phones to ensure good reception and transmission quality provided by MNOs.

In selecting the location, the downtown area of Mexico City stands out for its rich blend of economic, cultural, and social activities. A particularly strategic zone within this area is the public park of Alameda Central, enveloped by museums, theatres, hotels, offices, restaurants, and commercial stores.

This dynamic environment attracts many office workers, residents, tourists, and visitors participating in recreational activities, generating a significant demand for mobile data transfer.

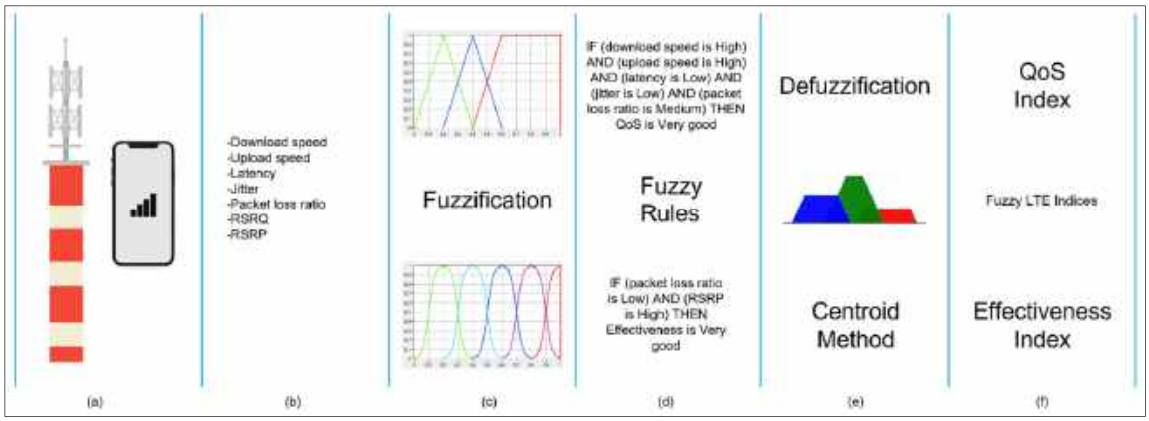



**Fig. 1.** Fuzzy model methodology: (a) information repository; (b) five QoS KPIs of data transfer and two LTE network signal parameters; (c) fuzzification process based on membership functions; (d) fuzzy rules (372); (e) defuzzification process with the centroid method; and (f) fuzzy QoS and effectiveness indices of the 4G LTE network

To assess the QoS of LTE network data transfer, we have considered the recommendations of prominent organisations such as ITU [9], the European Telecommunications Standards Institute (ETSI) [40], and the Body of European Regulators for Electronic Communications (BEREC) [41]. These guidelines indicate that the appropriate KPIs for this evaluation are download speed, upload speed, latency, jitter, and packet loss rate.

Notably, the packet loss rate is not explicitly included in the information repository but can be derived by applying Eq. (1), as both received and sent packets are available in the repository. Additionally, we propose incorporating the reference signal received quality (RSRQ) parameter to account for the transmission medium.

$$\text{packet loss rate} = \frac{\text{packets received}}{\text{packets sent}}\ [\%]. \tag{1}$$

Furthermore, we derive the data transfer effectiveness index to complement the evaluation of the LTE network service. The packet loss rate is the corresponding KPI for assessing this index. Moreover, as with the QoS, we introduce a signal parameter to consider the transmission medium: the reference signal received power (RSRP).

Ookla [42], the organization responsible for the crowdsourcing measurements, provided the data dictionary of the information repository available in Table A1. However, it is worth noting that the samples reported include outliers, which require a filtering or debugging process that we outline in Section 2.2.

### 2.2 Filtering the Information Repository

It is necessary to verify the data to ensure an accurate network performance evaluation, as mentioned in [43]. Each parameter has a specific valid range indicated in the data dictionary or the reports published by the company that carried out the measurements [44].

The mobile broadband service reports an average latency of 50 ms, with values ranging from 1 ms to 100 ms. Jitter has a maximum allowable value of 30 ms; any value higher than this is considered invalid. The packet loss rate should fall between 0% and 1%, and values outside this range are discarded. Likewise, the valid range for RSRP is from -120 dBm to -44 dBm, with -44 dBm being the maximum value. For RSRQ, the acceptable values range from -19.5 dB to -3 dB, with -19.5 dB being the minimum value.

For the download and upload speeds, we used the recommendations from the telecommunications regulatory body in Mexico, the Federal Telecommunications Institute [45], and the 2021 Mexico Median Country Speeds report [44]. These state that the valid range for download speed is 4 Mbps to 300 Mbps, and upload speed is 1 Mbps to 100 Mbps. Table 1 presents the range of values used as filters for each parameter.


**Fig. 2.** Map showing the geographical location where information was obtained. The colors indicate different network operators named with the following terminology: MNO 1 is AT&T, MNO 2 is Telcel, MNO 3 is Altan Redes, MNO 4 is Movistar, and MNO 5 is Unefón

Applying these filters to the information repository resulted in a reduction of data samples from 607 to 385, which were geolocated across 39 different locations, as seen in Fig. 2.

The latitude and longitude coordinates of these 39 geolocated points can be found in Table A2. Out of the 385 valid data samples, measurements are available for each day of the first two months of 2021, representing five MNOs. However, it is essential to note that not every georeferenced point contains data for each day and each operator. Considering this, membership functions for the fuzzy inference model are proposed and presented in Section 2.3.

### 2.3 Membership Functions: Input Variables

The fuzzy inference model evaluates seven input variables: five are linear data transfer QoS KPIs, while the other two are logarithmic signal parameters. Consequently, the membership functions for the linear variables are of triangular and trapezoidal types. On the other hand, the membership functions for the logarithmic variables are sigmoidal.

Eqs. (2) and (3) give the parameterisation for the triangular and trapezoidal membership functions [46]. The triangular membership function is defined by three parameters, $a$, $b$, and $c$, as follows:

$$\mu(x;a,b,c)=\begin{cases}0, & x\le a,\\ \dfrac{x-a}{b-a}, & a\le x\le b,\\ \dfrac{c-x}{c-b}, & b\le x\le c,\\ 0, & c\le x,\end{cases} \tag{2}$$

where the parameters $a$, $b$, and $c$, with $a<b<c$, determine the triangle's three corners. Meanwhile, the trapezoidal membership function is defined by four parameters, $a$, $b$, $c$, and $d$, as follows:

$$\mu(x;a,b,c,d)=\begin{cases}0, & x\le a,\\ \dfrac{x-a}{b-a}, & a\le x\le b,\\ 1, & b\le x\le c,\\ \dfrac{d-x}{d-c}, & c\le x\le d,\\ 0, & d\le x,\end{cases} \tag{3}$$

where the parameters $a$, $b$, $c$, and $d$, with $a<b\le c<d$, determine the value of $x$ for the four corners of the trapezoid. The input membership functions for the QoS KPIs, i.e., download speed, upload speed, latency, jitter, and packet loss rate, are depicted in Figs. 3–7.

These linear membership functions have been fitted within the ranges described in Section 2.2. Three linguistic values are considered for all input variables: Low, Medium, and High.

For instance, as depicted in Fig. 3, when the download speed falls from 0 to 4, it is classified under the linguistic value of Low. In the range of 4 to 38, the linguistic variable Medium exhibits a degree of membership that increases linearly with a high positive slope.

Similarly, Fig. 4 illustrates that upload speed follows a comparable pattern from 1 to 14.

However, this high positive slope is not observed for latency (1 to 50), jitter (0 to 15), and packet loss rate (0 to 0.005), as shown in Figs. 5-7, respectively. In these cases, the membership functions increase with a lower slope.

Conversely, for download and upload speeds starting from 38 and 14, respectively, the membership degree of the linguistic variable Medium decreases with a steeper negative slope than latency, jitter, and packet loss rate starting from 50, 15, and 0.005, respectively. Furthermore, the linguistic variable High reaches its maximum values for download speed, upload speed, and latency, beginning from 72, 29, and 99.

The sigmoidal membership function and its parameters are defined by Eq. (4):

$$\mu(x; a, c) = \frac{1}{1+e^{-a(x-c)}}. \quad (4)$$

Here, $a$ controls the slope at the crossing point $x = c$. As stated in [47], the sign of the parameter $a$ determines whether the sigmoidal membership function is intrinsically open to the right or left, making it suitable for representing concepts such as "very large" or “very negative”.

Fig. 8 illustrates the sigmoidal membership functions for the variables RSRP and RSRQ. Both variables were rescaled and normalised using the following expression (Eq. 5):

$$\frac{(x-x_{min})}{(x_{max}-x_{min})}, \quad (5)$$

where $x_{min}$ represents the minimum valid value of the variable to be normalised, $x_{max}$ is the maximum value of the variable, and $x$ is the variable to be normalised. For RSRP, the range of values to be normalised is from –120 dBm to –44 dBm, while for RSRQ, the range is from –19.5 dB to –3 dB. These sigmoidal membership functions were initially fitted with expert knowledge, followed by a manual adjustment.

**Table 1.** Filters for each data transfer QoS KPI or signal parameter considered

| Parameter | Unit | Filter |
|---|---|---|
| Latency | ms | 1 to 100 |
| Jitter | ms | 0 to 30 |
| Packet loss rate | % | 0 to 1 |
| RSRP | dBm | -120 to -44 |
| RSRQ | dB | -19.5 to -3 |
| Download speed | Mbps | 4 to 300 |
| Upload speed | Mbps | 1 to 100 |

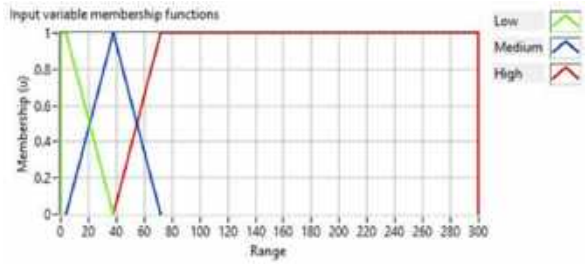



**Fig. 3.** Trapezoidal and triangular membership functions for download speed

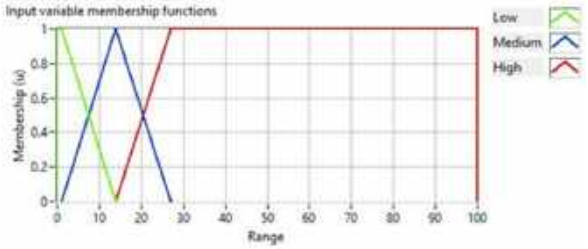



**Fig. 4.** Trapezoidal and triangular membership functions for upload speed

### 2.4 Membership Functions: Output Variables

The QoS and effectiveness indices are the output variables obtained from the fuzzy inference model. Fig. 9 illustrates sigmoid-shaped membership functions for both indices because each output is evaluated with at least one non-linear variable. Additionally, linguistic values such as Very poor,

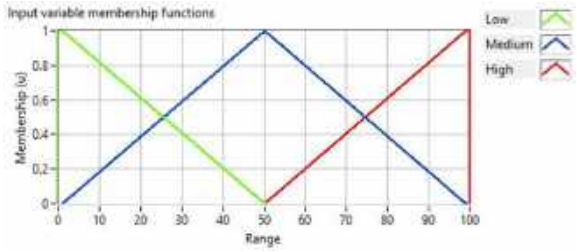



**Fig. 5.** Trapezoidal and triangular membership functions for latency

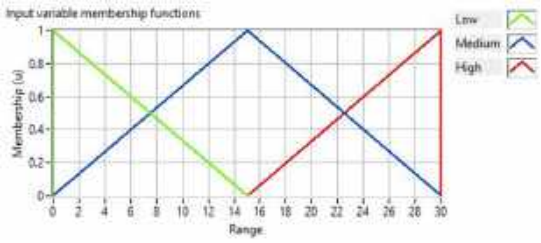



**Fig. 6.** Triangular membership functions for jitter

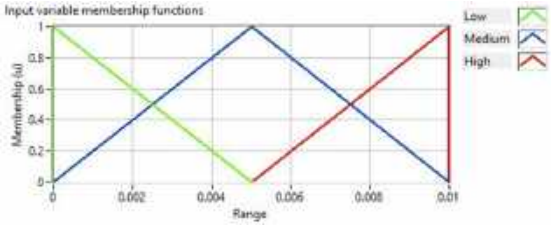



**Fig. 7.** Triangular membership functions for packet loss rate

Poor, Acceptable, Good, and Very good have been considered to establish a quality evaluation scale for both indices.

On the other hand, the fuzzy quality index is determined by evaluating the following input variables: download speed, upload speed, latency, jitter, packet loss rate, and RSRQ. This index depends on the values of the input variables. For instance, if the download speed, upload speed, and RSRQ have a High value, while the latency, jitter, and packet loss rate have a Low value, the resulting quality is classified as Very good.

Similarly, the fuzzy effectiveness index, which assesses packet loss rate and RSRP as input variables, will yield a Very good value when the packet loss rate is Low and the RSRP is High. The fuzzy rules that complement the membership functions are described in detail in Section 2.5.

### 2.5 Fuzzy Rules

Fuzzy rules of the type if–then were proposed using the AND (minimum) connector in the antecedent for both fuzzy indices. Each input variable is associated with three linguistic variables. The general expression for this set of variables is given by Eq. (6):

$$\mathrm{T}(\mathrm{variable}) = \{\mathrm{Low}, \mathrm{Medium}, \mathrm{High}\}, \tag{6}$$

where $\mathrm{T}$ represents the set of the variable in question (e.g., download speed, latency, etc.), and $\mathrm{Low}$, $\mathrm{Medium}$, and $\mathrm{High}$ are the linguistic variables in the set. The following expression was used to determine the number of rules:

$$\mathrm{Q} = \mathrm{M}^{\mathrm{N}}, \tag{7}$$

where $\mathrm{M}$ is the size of the linguistic variable set, $\mathrm{N}$ is the number of input variables, and $\mathrm{Q}$ is the number of rules.

First, we obtained the number of rules for the QoS fuzzy index by considering six input variables and three linguistic variables, resulting in 729 rules for QoS. Similarly, the effectiveness index yielded nine fuzzy rules after considering two input variables and three linguistic variables. Thus, the total number of rules for both indices is 738, encompassing all possible combinations of input variables using the AND connector.

We considered reducing the number of rules to avoid redundancy and computational cost associated with many of them. For this purpose, the rules were analysed from the point of view of expert knowledge in mobile networks. We proposed a weighting scheme for the input variables, giving higher importance to the KPIs variables relative to the signal parameter ones since regulatory bodies recommend KPIs for the QoS evaluation.

For the QoS case, where five of the six input variables are KPIs, and one is a signal parameter, a weighting of 7/36 was assigned to each of the five KPIs and 1/36 to RSRQ. The sum of the weights for all six variables is equal to 1, namely:

$$\frac{7}{36}+\frac{7}{36}+\frac{7}{36}+\frac{7}{36}+\frac{7}{36}+\frac{1}{36}=1. \tag{8}$$

Likewise, different weights were assigned to the input variables for the fuzzy effectiveness index. A weight of 3/4 was proposed for the KPI and 1/4 for


**Fig. 8.** Sigmoidal membership functions for the RSRP and RSRQ (the graphical representation is the same for both)

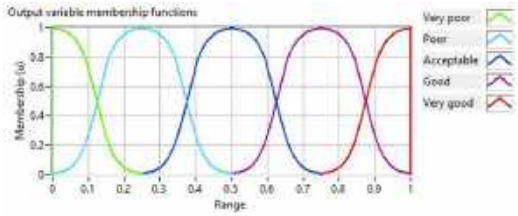



**Fig. 9.** Sigmoidal membership functions for the QoS and Effectiveness indices (the graphical representation is the same for both indices)

the RSRP. The sum of the weights for both variables is again equal to one.

After applying the weights mentioned earlier to the 738 rules, we found coincidences in two different types of groups, each containing three rules. In the first group, five of the six variables in the antecedent held the same value, and the consequent had the same result for all three rules. Consequently, these rules could be merged into one. Consider the following rules as an example:

$R_{L1}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **AND** (RSRQ is Low) **THEN** QoS is Very poor.

$R_{L2}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **AND** (RSRQ is Medium) **THEN** QoS is Very poor.

$R_{L3}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **AND** (RSRQ is High) **THEN** QoS is Very poor.

Observing rules $R_{L1}$, $R_{L2}$, and $R_{L3}$, we can see that the linguistic values for download speed (Low), upload speed (Low), latency (High), jitter (High), and packet loss ratio (High) are repeated in the antecedent. However, the RSRQ variable carries different linguistic values for each rule (Low, Medium, and High).

On the other hand, the consequent consistently yields the same value for QoS (Very poor). Since the RSRQ variable covers all three linguistic possibilities in this group without affecting the QoS result, we can merge these three rules into one. As a result, the RSRQ variable is removed, obtaining rule $R_1$:

$R_1$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **THEN** QoS is Very poor.

A similar situation occurs with the second group of rules shown below. In the antecedent, the variables maintaining the same linguistic value are download speed, upload speed, latency, jitter, and packet loss ratio, while the RSRQ variable varies its linguistic value. In the rule's consequent, the QoS has two different outcomes.

$R_{L25}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is Low) **THEN** QoS is Poor.

$R_{L26}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is Medium) **THEN** QoS is Acceptable.

$R_{L27}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is High) **THEN** QoS is Acceptable.

Here, two rules share the same linguistic value, while the third rule differs. Thus, we can merge two rules into one, reducing the number of rules from three to two. In the antecedent of rules $R_{L25}$, $R_{L26}$, and $R_{L27}$, we can observe that only the RSRQ variable changes its linguistic value among the three possibilities: Low, Medium, and High.

On the other hand, in the consequent, the QoS value remains "Acceptable" for rules $R_{L26}$ and $R_{L27}$, while for rule $R_{L25}$, it is classified as "Poor." Keeping this in mind, we merge rules $R_{L26}$ and $R_{L27}$ into rule $R_{13}$. However, it should be noted that to avoid affecting the result of this group, rule $R_{12}$ is placed before the merged rule $R_{13}$, as shown below:

$R_{12}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is

**Table 2.** File format for KPIs and signal parameter data in the information repository

| Download Speed (Mbps) | Upload Speed (Mbps) | Latency (ms) | Jitter (ms) | Packet Loss Rate (%) | RSRP | RSRQ |
|---|---|---|---|---|---|---|
| 21.92 | 26.436 | 31 | 9.2 | 0 | 0.36 | 0.70588235 |
| 67.943 | 27.148 | 32 | 7.8 | 0 | 0.36 | 0.70588235 |

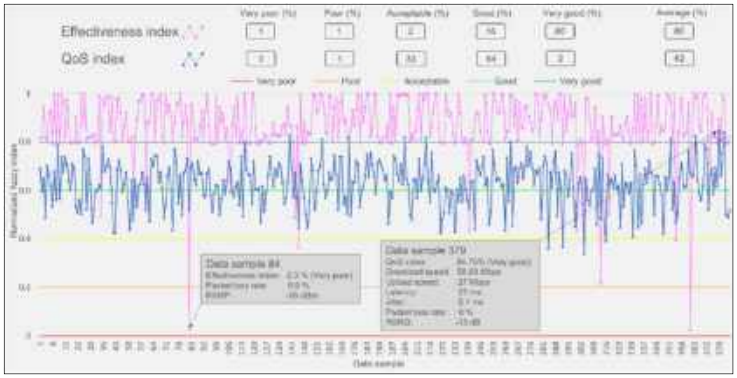



**Fig. 10.** QoS and effectiveness fuzzy indices for the 385 data samples evaluated in the study zone

Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is Low) **THEN** QoS is Poor.

$R_{13}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **THEN** QoS is Acceptable.

Consequently, after merging sets of three rules, the total number of rules was reduced from 738 to 372. Some representative rules are presented below. A summary of these rules can be found in Table A3.

$R_1$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **THEN** QoS is Very poor.

$R_{12}$: **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is Low) **THEN** QoS is Poor.

$R_{182}$: **IF** (download speed is Medium) **AND** (upload speed is Medium) **AND** (latency is Medium) **AND** (jitter is Medium) **AND** (packet loss ratio is Medium) **THEN** QoS is Acceptable.

$R_{351}$: **IF** (download speed is High) **AND** (upload speed is High) **AND** (latency is Low) **AND** (jitter is High) **AND** (packet loss ratio is High) **AND** (RSRQ is High) **THEN** QoS is Good.

$R_{363}$: **IF** (download speed is High) **AND** (upload speed is High) **AND** (latency is Low) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **THEN** QoS is Very good.

$R_{364}$: **IF** (packet loss ratio is High) **AND** (RSRP is Low) **THEN** Effectiveness is Very poor.

$R_{366}$: **IF** (packet loss ratio is High) **AND** (RSRP is High) **THEN** Effectiveness is Poor.

$R_{368}$: **IF** (packet loss ratio is Medium) **AND** (RSRP is Medium) **THEN** Effectiveness is Acceptable.

$R_{370}$: **IF** (packet loss ratio is Low) **AND** (RSRP is Low) **THEN** Effectiveness is Good.

$R_{372}$: **IF** (packet loss ratio is Low) **AND** (RSRP is High) **THEN** Effectiveness is Very good.

In rule $R_1$, when there is poor data download and upload speed performance, high latency, jitter, and packet loss, the QoS is classified as "Very poor." On the other hand, for rule $R_{12}$, the QoS is "Poor" when the download speed is less than 21 Mbps, the upload speed is less than 7.5 Mbps, latency is between 75 ms and 100 ms, jitter is between 22.5 ms and 30 ms, packet loss ratio is between 0.25% and 0.75%, and the RSRQ is

**Table 3.** Fuzzy indices range corresponding to linguistic variables

| Fuzzy QoS Index × $10^2$ (%) | Fuzzy Effectiveness Index × $10^2$ (%) | Linguistic variables |
|---|---|---|
| 0 to 0.2 | 0 to 0.2 | Very poor |
| 0.2 to 0.4 | 0.2 to 0.4 | Poor |
| 0.4 to 0.6 | 0.4 to 0.6 | Acceptable |
| 0.6 to 0.8 | 0.6 to 0.8 | Good |
| 0.8 to 1 | 0.8 to 1 | Very good |

**Table 4.** Input and output variables used in evaluating QoS for the area of study

| Data Sample | Download Speed (Mbps) | Upload Speed (Mbps) | Latency (ms) | Jitter (ms) | Packet Loss Rate (%) | RSRQ (dB) | QoS ×$10^2$ (%) |
|---|---|---|---|---|---|---|---|
| 5 | 38.928 | 23.443 | 37 | 6.6 | 0 | –11 | 0.700233 |
| 33 | 23.104 | 6.268 | 50 | 8.9 | 0 | –13 | 0.600372 |
| 149 | 9.383 | 1.23 | 41 | 12.6 | 0 | –14 | 0.5004 |
| 158 | 36.921 | 41.222 | 16 | 9.1 | 0 | –15 | 0.810547 |
| 304 | 4.474 | 4.241 | 21 | 19.3 | 0.003597122 | –17 | 0.337567 |
| 331 | 4.643 | 10.326 | 71 | 15.7 | 0 | –8 | 0.408525 |
| 379 | 56.694 | 27 | 21 | 15.1 | 0 | –13 | 0.847512 |
| 157 | 0 | –81 | 0.99994181 | | | | |
| 300 | 0.003355705 | –109 | 0.43180837 | | | | |
| 374 | 0.003355705 | –77 | 0.60668714 | | | | |

**Table 5.** Input and output variables used in the evaluation of effectiveness for the area of study

| Data Sample | Packet Loss Rate × $10^2$ (%) | RSRP (dBm) | Effectiveness × $10^2$ (%) |
|---|---|---|---|
| 19 | 0 | –108 | 0.7983722 |
| 60 | 0.003355705 | –97 | 0.54166604 |
| 84 | 0.009950249 | –95 | 0.02383161 |
| 145 | 0.003937008 | –115 | 0.36244197 |

higher than –7.125 dB (i.e., Low download and upload speeds, High latency, jitter, and RSRQ; Medium packet loss ratio).

Rule $R_{182}$ demonstrates that QoS is "Acceptable" when each variable fall within their medium range, i.e., download speed between 21 Mbps and 55 Mbps, upload speed between 7.5 Mbps and 20.5 Mbps, latency between 25 ms and 75 ms, jitter between 7.5 ms and 22.5 ms, packet loss ratio between 0.25% and 0.75%, and RSRQ

**Table 6.** Distribution of QoS by operator in the study area

| QoS | MNO 1 (%) | MNO 2 (%) | MNO 3 (%) | MNO 4 (%) | MNO 5 (%) |
|---|---|---|---|---|---|
| Very poor | 0 | 0 | 0 | 0 | 0 |
| Poor | 0 | 2 | 0 | 0 | 0 |
| Acceptable | 36 | 35 | 27 | 45 | 0 |
| Good | 62 | 60 | 71 | 55 | 86 |
| Very good | 2 | 3 | 2 | 0 | 14 |
| Average | 64 | 61 | 65 | 58 | 68 |

**Table 7.** Distribution of service effectiveness by operator in the study area

| Effectiveness | MNO 1 (%) | MNO 2 (%) | MNO 3 (%) | MNO 4 (%) | MNO 5 (%) |
|---|---|---|---|---|---|
| Very poor | 0 | 1 | 0 | 0 | 0 |
| Poor | 0 | 2 | 0 | 0 | 0 |
| Acceptable | 0 | 5 | 0 | 0 | 0 |
| Good | 8 | 20 | 14 | 18 | 0 |
| Very good | 92 | 72 | 86 | 82 | 100 |
| Average | 90 | 84 | 89 | 88 | 99 |

between –15.375 dB and –7.125 dB (all variables have a value of Medium). Rule $R_{351}$ states that if high values are observed for download speed (55 Mbps) and upload speed (20.5 Mbps) and low values for latency (25 ms), jitter (7.5 ms), packet loss ratio (0.25%), and RSRQ (–15.375 dB) (i.e., High download and upload speeds, Low latency, jitter, and packet loss ratio), then the QoS is classified as "Good".

Similarly, rule $R_{363}$ states that the QoS is classified as "Very Good" when the download and upload speeds are higher than 55 Mbps and 20.5 Mbps, respectively. The latency, jitter, and packet loss ratio are lower than 25 ms, 7.5 ms and 0.25%, respectively (i.e., High download and upload speeds, Low latency, jitter, and packet loss ratio).

If the packet loss rate exceeds 0.75% and the RSRP values are below –101 dBm (packet loss rate is High and RSRP is Low), then the effectiveness is categorised as "Very poor" in rule $R_{364}$. Rule $R_{366}$ classifies the effectiveness as "Poor" if the packet loss rate is higher than 0.75% and the RSRP is –63 dBm (packet loss rate and RSRP are both High).

The effectiveness is considered "Acceptable" in rule $R_{368}$ when the packet loss rate falls between 0.25% and 0.75%, and the RSRP is between –110 dBm and –63 dBm (packet loss rate and RSRP are both Medium).

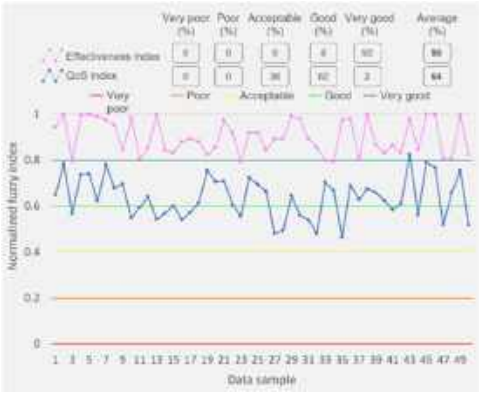
If the packet loss rate and RSRP are below 0.25% and –101 dBm, respectively (Low packet loss rate and RSRP), then the effectiveness is categorised as "Good" in rule $R_{370}$. Similarly, in rule $R_{372}$, if the data transfer is optimal with a packet loss rate below 0.25% and RSRP greater than –63 dBm (Low packet loss rate and High RSRP), the effectiveness is classified as "Very good."

## 3 Results

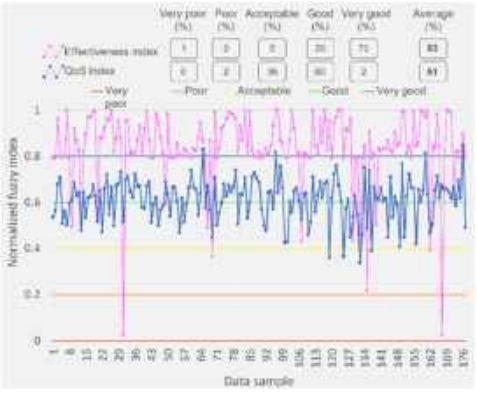
The fuzzy inference model algorithm was developed and programmed in software with specialised fuzzy logic libraries. The program

ISSN 2007-9737

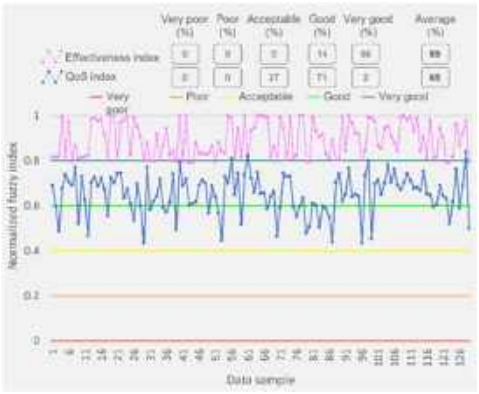




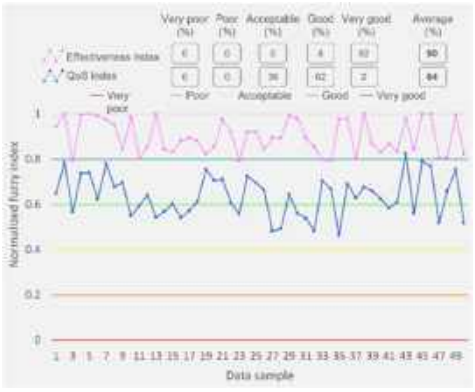

a)



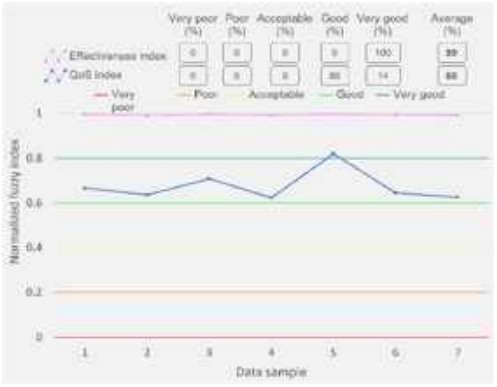

b)




c)




d)




e)

**Fig. 11.** Fuzzy QoS and effectiveness indices obtained by the mobile network operators (MNO): (a) MNO 1, (b) MNO 2, (c) MNO 3, (d) MNO 4, and (e) MNO 5

allows for manual or automatic evaluation of the input variables through the following steps:

1. Select the manual or automatic evaluation mode for the data rows.

**Table 8.** Fuzzy QoS index for MNO 1 by location

| No. | Location no. | Latitude | Longitude | MNO | QoS Index | Category |
|---|---|---|---|---|---|---|
| 1 | 3 | 19.434 | –99.149 | 1 | 0.64829325 | Good |
| 2 | 9 | 19.434 | –99.147 | 1 | 0.67515632 | Good |
| 3 | 14 | 19.435 | –99.145 | 1 | 0.70858523 | Good |
| 4 | 17 | 19.436 | –99.144 | 1 | 0.57596928 | Acceptable |
| 5 | 18 | 19.435 | –99.144 | 1 | 0.61902345 | Good |
| 6 | 19 | 19.433 | –99.144 | 1 | 0.66704351 | Good |
| 7 | 20 | 19.438 | –99.142 | 1 | 0.62331306 | Good |
| 8 | 21 | 19.434 | –99.142 | 1 | 0.58415889 | Acceptable |
| 9 | 26 | 19.435 | –99.141 | 1 | 0.61081482 | Good |
| 10 | 30 | 19.434 | –99.139 | 1 | 0.82497577 | Very good |
| 11 | 31 | 19.432 | –99.139 | 1 | 0.55999562 | Acceptable |
| 12 | 32 | 19.434 | –99.138 | 1 | 0.77987155 | Good |
| 13 | 33 | 19.433 | –99.138 | 1 | 0.51886548 | Acceptable |
| 14 | 35 | 19.435 | –99.137 | 1 | 0.65858975 | Good |
| 15 | 36 | 19.433 | –99.137 | 1 | 0.75429903 | Good |
| 16 | 37 | 19.435 | –99.136 | 1 | 0.51764475 | Acceptable |

2. Prompt the user to provide a plain text file containing the information repository data. This file comprises seven columns, each representing a KPI or a received signal parameter.
3. Request the file containing the fuzzy rules.
4. Evaluate each row from the repository data file using the Mamdani fuzzy method.
5. If the evaluation mode is manual, proceed to the next step. Otherwise, return to step 4 until all rows in the repository data file have been evaluated.
6. Calculate the fuzzy quality and effectiveness indices.

The structure of the information repository file should follow the format presented in Table 2. This file should not contain headers, only the seven-column data representing the five KPIs and two signal parameters.

The proposed fuzzy inference model employs seven LTE network parameters to assess 4G data transmission quality and efficiency. The two descriptive indices summarise the percentage of QoS and the percentage of effectiveness, allowing even users without technical expertise to evaluate and interpret the results quickly.

However, it is essential to note that while LTE network indices provide an interpretable measure of network QoS, they do not replace the KPIs or parameters recommended by regulators, which are the basis of the fuzzy model.

We use a dataset of 385 samples from the central Alameda zone of Mexico City to evaluate the model. The input variables for the fuzzy quality index are download speed, upload speed, latency, jitter, packet loss rate, and RSRQ.

On the other hand, the variables for the effectiveness index are packet loss rate and RSRP.


**Fig. 12.** Thematic map of the data transfer QoS index for MNO 1


**Fig. 13.** Thematic map of the effectiveness index for MNO 1

To facilitate interpretation, the values obtained for both fuzzy indices are normalised and classified into five levels associated with the five linguistic variables used in the fuzzy rules: Very poor, Poor, Acceptable, Good, and Very good. The normalised value is divided into five equal sections of size 0.2, each representing a different linguistic variable.

For instance, the range from 0 to 0.2 corresponds to the "Very poor" category, while the range from 0.2 to 0.4 means "Poor". Similarly, the range from 0.4 to 0.6 indicates "Acceptable, the range from 0.6 to 0.8 is "Good", and the range from 0.8 to 1 represents "Very good". In other words, the numerical values can be represented by qualitative variables, as summarised in Table 3, which reflect the QoS perceived by the user QoE.

Fig. 10 gives a visual representation of the results for both fuzzy indices. The horizontal lines highlight the minimum value for each linguistic variable category.

For example, the horizontal axis appears in red at the 0 level to indicate the minimum value for the "Very poor" category. Similarly, the horizontal axis is orange, yellow, green, and dark green at the 0.2, 0.4, 0.6, and 0.8 index levels representing the minimum values for the "Poor", "Acceptable", "Good", and "Very good" categories, respectively.

**Table A1.** The telecommunications department of IPN ESIME Zacatenco provided the information repository

| Field Name in Extract | Definition | Data type | Max Field Size |
|---|---|---|---|
| test_id | A unique ID for every speed test performed based on the platform. | number | 11 |
| test_date | Date and time of the test in UTC (default). Previously, this data was presented in Pacific time (default). | datetime | 19 |
| download_kbps | The result of the download portion of the test was measured in kilobits per second. | number | 11 |
| upload_kbps | The result of the upload portion of the test was measured in kilobits per second. | number | 11 |
| latency | The result of the latency portion of the test was measured in milliseconds. | number | 6 |
| client_city | The city where the test was conducted. If this field is labelled unknown and the location type field equals 2, we cannot confidently assign a city to the record, and it is only accurate to the region level. | text | 255 |
| client_latitude | The latitude of the device conducting the test. If the location type equals 2, this value is approximate. | number | 10 |
| client_longitude | The longitude of the device conducting the test. If the location type equals 2, this value is approximate. | number | 10 |
| network_operator_name | The device displays the name of the network operator. This field is generally accurate but is not reliable. Custom versions of Android, including carrier-specific versions, may modify this string, and carriers have used different spellings and brandings on their network names. 2020-05-28: When Ookla can more accurately identify the active network operator via Android CellIdentity APIs, it will overwrite the network_operator_name based on Ookla-maintained MCCMNC-to-network relations. | text | 255 |
| MCC | A Mobile Country Code (MCC) is a three-digit code specific to the country where the network operator is located. A list of codes and countries can be found here. | number | 3 |
| mnc | Mobile Network Code (MNC). A three-digit identifier is specific to a network operator within a given country. Each mobile network can be uniquely identified when combined with the Mobile Country Code. A comprehensive list of Mobile Network Codes can be found here, but each country regulates their MNCs individually so that codes may appear or change without any notice. | number | 3 |
| pre_connection_type | A number representing the device's connection type as detected before the test begins. Please see the connection type table for an explanation of each type. Connection type as detected before the test starts: 0 = Unknown,1 = Cell, 2 = Wi-Fi, 3 = GPRS, 4 = EDGE, 5 = UMTS, 6 = CDMA, 7 = EVDO0, 8 = EVDOA, 9 = OnexRTT, 10 = HSDPA, 11 = HSPA, 12 = IDEN, 13 = EHRPD, 14 = EVDOB, 15 = LTE, 16 = HSUPA, 17 = HSPAP, 18 = GSM, 19 = TDSCDMA, 20 = IWLAN, 21 = LTE-CA, 22 = Ethernet, 23 = Bluetooth, 24 = NR | number | 4 |
| post_connection_type | A number representing the device's connection type as detected at the end of the test. Please see the connection type table for an explanation of each type. Connection type as witnessed at the end of the test: 0 = Unknown,1 = Cell, 2 = Wi-Fi, 3 = GPRS, 4 = EDGE, 5 = UMTS, 6 = CDMA, 7 = EVDO0, 8 = EVDOA, 9 = OnexRTT, 10 = HSDPA, 11 = HSPA, 12 = IDEN, 13 = EHRPD, 14 = EVDOB, 15 = LTE, 16 = HSUPA, 17 = HSPAP, 18 = GSM, 19 = TDSCDMA, 20 = IWLAN, 21 = LTE-CA, 22 = Ethernet, 23 = Bluetooth, 24 = NR | number | 4 |
| brand | The consumer-facing brand of the device. | text | 255 |
| manufacturer | The device manufacturer. | text | 255 |
| model | User device's model. | text | 255 |
| jitter_a | The latency variance over time determines the internet connection's stability. | number | 4 |

Thus, if the index is above the minimum value of one category and below the next, it will correspond to the lower category.

For example, consider the effectiveness index result for sample 84, the lowest value depicted in Fig. 10. Although it surpasses the minimum of the

| | | | |
|---|---|---|---|
| ploss_sent_a | The number of packages sent to the host server from the device. To determine packet loss percentage, take the ploss_recv divided by ploss_sent. 100% means zero packet loss since 100% of the packets were sent and received. | number | 5 |
| ploss_recv_a | The number of packages received by the host server from the device. To determine packet loss percentage, take the ploss_recv divided by ploss_sent. 100% means zero packet loss since 100% of the packets were sent and received. | number | 5 |
| tr_latency_a | The time is taken for the packet to be received after the first hop in the traceroute. | Number | 5 |
| tr_ip_1_a | This is the IP address of the second hop of the traceroute during a test. | text | 15 |
| earfcn_a | EARFCN stands for E-UTRA Absolute Radio Frequency Channel Number. In LTE, the carrier frequency in the uplink and downlink is designated by EARFCN, which ranges between 0 and 65535. EARFCN uniquely identify the LTE band and carrier frequency. ... EARFCN is independent of channel bandwidth. | number | 8 |
| rsrp_a | Reference Signal Received Power. LTE metric displaying the received power of the reference LTE signal, similar to the old school "signal strength". Range: -40 to -140, where -140 is the worst and -44 is the highest the device will report | number | 4 |
| rsrq_a | Reference Signal Received Quality The received quality of the LTE reference signal. Range: -19.5 to -3, where -3 is best | number | 5 |
| rssnr_a | Reference Signal Signal-to-Noise Ratio is Perhaps the most important KPI, but it needs to be reported more adequately in Android OS. It's the ratio between the noise and signal of the LTE data transmission. The range is -30 to +30, where +30 is best | number | 3 |
| cqi_a | CQI stands for Channel Quality Indicator. As the name implies, it is an indicator carrying the information on how good/bad the communication channel quality is. LTE has 15 different CQI values ranging from 1 to 15 and mapping between CQI and modulation scheme. In HSDPA, the CQI value ranges from 0 to 30. 30 indicates the best channel quality, and 0,1 tells the poorest channel quality. | number | 2 |
| cellbandwidth_a | Cell bandwidth in kHz | number | 6 |

**Table A2.** Geographic coordinates of the 39 points where the information repository is distributed

| No. | Latitude | Longitude | No. | Latitude | Longitude | No. | Latitude | Longitude |
|---|---|---|---|---|---|---|---|---|
| 1 | 19.438 | –99.149 | 14 | 19.435 | –99.145 | 27 | 19.434 | –99.141 |
| 2 | 19.436 | –99.149 | 15 | 19.438 | –99.144 | 28 | 19.433 | –99.141 |
| 3 | 19.434 | –99.149 | 16 | 19.437 | –99.144 | 29 | 19.435 | –99.14 |
| 4 | 19.437 | –99.148 | 17 | 19.436 | –99.144 | 30 | 19.434 | –99.139 |
| 5 | 19.436 | –99.148 | 18 | 19.435 | –99.144 | 31 | 19.432 | –99.139 |
| 6 | 19.435 | –99.148 | 19 | 19.433 | –99.144 | 32 | 19.434 | –99.138 |
| 7 | 19.432 | –99.148 | 20 | 19.438 | –99.142 | 33 | 19.433 | –99.138 |
| 8 | 19.437 | –99.147 | 21 | 19.434 | –99.142 | 34 | 19.438 | –99.137 |
| 9 | 19.434 | –99.147 | 22 | 19.433 | –99.142 | 35 | 19.435 | –99.137 |
| 10 | 19.433 | –99.147 | 23 | 19.438 | –99.141 | 36 | 19.433 | –99.137 |
| 11 | 19.438 | –99.146 | 24 | 19.437 | –99.141 | 37 | 19.435 | –99.136 |
| 12 | 19.433 | –99.146 | 25 | 19.436 | –99.141 | 38 | 19.434 | –99.136 |
| 13 | 19.432 | –99.146 | 26 | 19.435 | –99.141 | 39 | 19.432 | –99.136 |

"Very poor" category, it is below the minimum of the "Poor" category. Hence, this value is classified as "Very poor."

On the contrary, the QoS index result for data sample 379, the highest value depicted in Fig. 10, is 84.7512%.

This index considers the download speed, upload speed, latency, jitter, packet loss rate, and RSRQ, with values of 56.69 Mbps, 27 Mbps, 21 ms, 5.1 ms, 0%, and –13 dB, respectively.

Furthermore, Fig. 10 illustrates that the resulting values for QoS are distributed across four of the five categories.

Most values fall into the "Good" category at 64%, followed by 33% in the "Acceptable" category.

**Table A3.** Most relevant fuzzy rules to obtain the fuzzy indices of quality and effectiveness

| No. | Fuzzy Rules |
|---|---|
| 1 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is High) **THEN** QoS is Very poor. |
| 2 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is Medium) **THEN** QoS is Very poor. |
| 3 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is Low) **AND** (RSRQ is Low) **THEN** QoS is Very poor. |
| 4 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is High) **AND** (packet loss ratio is Low) **THEN** QoS is Poor. |
| ⋮ | ⋮ |
| 12 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **AND** (RSRQ is Low) **THEN** QoS is Poor. |
| 13 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is High) **AND** (jitter is Low) **AND** (packet loss ratio is Low) **THEN** QoS is Acceptable. |
| 14 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is Medium) **AND** (jitter is High) **AND** (packet loss ratio is High) **THEN** QoS is Very poor. |
| 15 | **IF** (download speed is Low) **AND** (upload speed is Low) **AND** (latency is Medium) **AND** (jitter is High) **AND** (packet loss ratio is Medium) **AND** (RSRQ is Low) **THEN** QoS is Very poor. |
| ⋮ | ⋮ |
| 182 | **IF (download speed is Medium) AND (upload speed is Medium) AND (latency is Medium) AND (jitter is Medium) AND (packet loss ratio is Medium) THEN QoS is Acceptable.** |
| 183 | **IF (download speed is Medium) AND (upload speed is Medium) AND (latency is Medium) AND (jitter is Medium) AND (packet loss ratio is Low) AND (RSRQ is High) THEN QoS is Good.** |
| 184 | **IF (download speed is Medium) AND (upload speed is Medium) AND (latency is Medium) AND (jitter is Medium) AND (packet loss ratio is Low) THEN QoS is Acceptable.** |
| 185 | **IF (download speed is Medium) AND (upload speed is Medium) AND (latency is Medium) AND (jitter is Low) AND (packet loss ratio is High) THEN QoS is Acceptable.** |
| ⋮ | ⋮ |
| 351 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is High) AND (packet loss ratio is High) AND (RSRQ is High) THEN QoS is Good.** |
| 352 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is High) AND (packet loss ratio is High) THEN QoS is Acceptable.** |
| 353 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is High) AND (packet loss ratio is Medium) THEN QoS is Good.** |
| 354 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is High) AND (packet loss ratio is Low) AND (RSRQ is High) THEN QoS is Very good.** |
| ⋮ | ⋮ |
| 360 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is Low) AND (packet loss ratio is High) AND (RSRQ is High) THEN QoS is Very good.** |
| 361 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is Low) AND (packet loss ratio is High) THEN QoS is Good.** |
| 362 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is Low) AND (packet loss ratio is Medium) THEN QoS is Very good.** |
| 363 | **IF (download speed is High) AND (upload speed is High) AND (latency is Low) AND (jitter is Low) AND (packet loss ratio is Low) THEN QoS is Very good.** |
| 364 | **IF (packet loss ratio is High) AND (RSRP is Low) THEN Effectiveness is Very poor.** |
| 365 | **IF (packet loss ratio is High) AND (RSRP is Medium) THEN Effectiveness is Very poor.** |
| 366 | **IF (packet loss ratio is High) AND (RSRP is High) THEN Effectiveness is Poor.** |
| 367 | **IF (packet loss ratio is Medium) AND (RSRP is Low) THEN Effectiveness is Poor.** |
| 368 | **IF (packet loss ratio is Medium) AND (RSRP is Medium) THEN Effectiveness is Acceptable.** |
| 369 | **IF (packet loss ratio is Medium) AND (RSRP is High) THEN Effectiveness is Good.** |
| 370 | **IF (packet loss ratio is Low) AND (RSRP is Low) THEN Effectiveness is Good.** |
| 371 | **IF (packet loss ratio is Low) AND (RSRP is Medium) THEN Effectiveness is Very good.** |
| 372 | **IF (packet loss ratio is Low) AND (RSRP is High) THEN Effectiveness is Very good.** |

The "Very good" category represents 2% of the values, while the "Poor" category accounts for only 1%. Notably, there is no distribution in the "Very poor" category.

As a result, the average QoS is 62%, indicating a “Good” QoS in the central Alameda zone of Mexico City. Table 4 displays the correlation between the input parameters and the resulting index.

For example, for data sample 304, the obtained QoS was classified as "Poor" at 31.9078%. This low rating is due to shared values in input

variables, including download speed (4.74 Mbps), upload speed (4.241 Mbps), latency (21 ms), and RSRQ (–17 dB). Low values for download speed, upload speed, and RSRQ worsen QoS, while a low value for latency improves QoS. Additionally, the jitter and packet loss rates have medium values of 19.3 ms and 33.7567%, respectively, which contribute to a QoS score of 50%. Thus, after processing all this information using the fuzzy model, sample 304 is classified as “Poor.”

Furthermore, Fig. 10 shows the results of the effectiveness index, with an average score of 82.1% classified as "Very good." The distribution of values across categories is as follows: "Very good" (80%), "Good" (16%), "Acceptable" (2%), "Poor" (1%), and "Very poor" (1%).

Additional details about the correlation between input variables and the effectiveness index can be found in Table 5. For instance, in the evaluation of sample 157, the resulting effectiveness is 99.994181%, classified as "Very good" due to a 0% packet loss rate and an average RSRP of –81 dBm.

The fuzzy indices categorised by MNO are displayed in Fig. 11 for the study area. The QoS index has a "Good" category for four out of the five MNOs (1, 2, 3, and 5), while MNO 4 received an "Acceptable" rating, as shown in Figs. 11(a), 11(b), 11(c), and 11(e) respectively. This indicates that, in this geographical location, MNOs 1, 2, 3, and 5 maintain a good quality of data transfer service. Meanwhile, MNO 4 lags two points below the quality of the other MNOs.

The average scores are 68%, 65%, 64%, 61%, and 58% for MNOs 5, 3, 1, 2, and 4, respectively. In summary, MNO 5 performs the best in terms of QoS, while MNO 4 has the lowest quality, as presented in Table 6. Moreover, all five operators achieved the "Very good" category in the fuzzy effectiveness index, indicating excellent effectiveness in data transmission in this geographical zone. The average scores for the MNOs 1, 2, 3, 4, and 5 are 90%, 84%, 89%, 88%, and 99%, respectively, as detailed in Table 7 and depicted in Figs. 11(a), 11(b), 11(c), 11(d), and (e).

The fuzzy indices for each MNO are presented as thematic maps below to provide users and regulatory bodies with easily understandable information. The thematic map in Fig. 12 illustrates the QoS for MNO 1, displaying data for 16 of the 39 georeferenced points in the repository. Among these points, ten are categorised as "Good" quality, five as "Acceptable," and one as "Very good." Importantly, no points are classified as "Poor" or "Very poor." Points that are not shown lack measurements for MNO 1. For a summary of these findings, please refer to Table 8.

Furthermore, Fig. 13 demonstrates that in all the 16 locations examined, the effectiveness category is consistently classified as "Very good." It is important to note that this georeferenced analysis was conducted for each MNO. However, as this work focuses on evaluating the LTE mobile network rather than the individual MNOs, the specific results for the other MNOs are not included.

## 4 Discussion

Numerous studies have examined QoS evaluation with diverse objectives. In [13], the relationship between KPIs and received signal parameters is investigated, although the aim is not to provide a joint index for both parameters.

In [14], the QoS priority is adjusted for some users to evaluate the scheduling of base stations. In addition, several works propose improvements in network resource planning to enhance QoS and user-perceived quality by increasing spectral efficiency [7. 15, 16, 17] or by analysing various configurations and parameters during video transmissions over the mobile network [19].Nonetheless, their objective is primarily to improve the infrastructure of MNOs.

On the other hand, [10, 24, 47] conduct statistical analyses of mobile network QoS, presenting distributions or estimates for missing values, but their approaches require substantially large data samples. Computational models have also been applied, employing various algorithms to optimise base station antenna tilt [25], optimise downlink resource allocation [26, 27], and maximise QoE [23]. Despite their aim to improve QoS and QoE, these proposals do not deliver relevant information to users.

Fuzzy systems have also been explored for LTE networks. [36, 37] assess QoS using Sugeno and ANFIS fuzzy models, yet they do not use

measurement data or provide understandable information to users about LTE network QoS.

Following ITU recommendations, this study combines KPIs and signal parameters to evaluate fuzzy indices of QoS and effectiveness in the 4G LTE mobile network. While the proposed fuzzy model can handle any dataset, even if appropriate adjustments are made, it could evaluate data from the 5G network; the results presented here are based on the information repository acquired in the central Alameda zone of Mexico City. This work offers the following contributions:

– Using crowdsourcing measurements allowed for information to be collected by end users through their devices, effectively integrating their perspective on network performance.
– Analyzing the information repository enabled determining the number of input variables and their combinations for each fuzzy index. This process led to designing a fuzzy model, employing membership functions and fuzzy rules that utilize KPIs and signal parameters to derive two fuzzy indexes.
– This fuzzy inference model effectively utilises Telecom expert knowledge and provides accurate evaluations using a small data sample. Its if–then rule-based inference enables easier comprehension of results for experts and users alike, as the indices are classified into five levels.
– Classifying results into these five levels allows for a qualitative understanding of current service quality and effectiveness at specific georeferenced points or within the study area. This aids experts and users in interpreting the mobile network service from a different perspective. Moreover, network operators can share these results to improve transparency.

## 5 Conclusion

This work proposes a fuzzy inference model to quantify seven LTE network parameters during data transfer using an information repository obtained through crowdsourcing. The ITU recommends specific KPIs and signal parameters for this purpose. We evaluated 385 data samples collected at 39 georeferenced points in the central Alameda zone of Mexico City.

Our model was designed based on the if–then rules derived from analysing the information repository and expert reasoning. As a result, we obtained fuzzy indices for data transfer QoS and effectiveness in the 4G LTE mobile network. This approach addressed the challenge of evaluating the network's QoS with a limited data sample, thanks to the flexibility offered by fuzzy models.

It is essential to clarify that our proposed fuzzy indices are not intended to replace existing descriptors. However, our developed fuzzy inference model brings two significant advantages compared to other works mentioned: a) it requires a smaller data sample to analyse the 4G LTE mobile network, and b) the evaluation results provide regulatory bodies and users with valuable insights into the network's quality. Finally, the realised Fuzzy Inference System can evaluate data from 5G networks, making the corresponding adaptions for this network.

As a part of future work to enhance the robustness of this research, we intend to consider utilising a more extensive information repository with additional data. This includes extending the study duration within the same geographic area for a more comprehensive analysis. Additionally, we plan to enhance the fuzzy model by incorporating an analytic hierarchy process and data for 5G networks.

## Acknowledgments

This work was supported financially by the National Polytechnic Institute (Instituto Politécnico Nacional, México) and the Mexican Council for Humanities, Science and Technology (CONAHCYT). We want to thank the Telecommunications Engineering branch of the Postgraduate Sciences program (PCIT), part of the Postgraduate Studies and Research Section (SEPI) at the Higher School of Mechanical and Electrical Engineering (ESIME), Zacatenco unit.

## References

1. **Lizunovs, A., Stafecka, A., Bobrovs, V. (2019).** Internet access service QoS and signal parameter measurements in urban

environment. 2019 23rd International Conference Electronics, IEEE. pp. 1–5. DOI: 10.1109/ELECTRONICS.2019.8765584.

2. **Stafecka, A., Lizunovs, A., Bobrovs, V., Gavars, P., Zariņš, Z. (2019).** Quality of service and signal evaluation parameter comparison between different mobile network operators in urban area. 2019 PhotonIcs & Electromagnetics Research Symposium-Spring, IEEE, pp. 3887–3894. DOI: 10.1109/PIERS-Spring46901.2019.9017741.
3. **Khan, S. A., Shayea, I., Ergen, M., Mohamad, H. (2022).** Handover management over dual connectivity in 5G technology with future ultra-dense mobile heterogeneous networks: A review. Engineering Science and Technology, an International Journal, Vol. 35. DOI: 10.1016/J.JESTCH.2022.101172.
4. **Yu, C., Chen, S., Wang, F., Wei, Z. (2021).** Improving 4G/5G air interface security: A survey of existing attacks on different LTE layers. Computer Networks, Vol. 201, pp. 108532. DOI: 10.1016/J.COMNET.2021.108532.
5. **Kelechi, A. H., Samson, U. A., Simeon, M., Obinna, O., Alex, A., Atayero, A. A. (2021).** The quality of service of the deployed LTE technology by mobile network operators in Abuja-Nigeria. International Journal of Electrical and Computer Engineering, Vol. 11, No. 3, pp. 2191.
6. **Sankaranarayanan, P., Gayathri, R., Tamijetchelvy, R. (2021)**. An MIH-enhanced fully distributed mobility management (MF-DMM) solution for real and non-real time CVBR traffic classes in mobile internet. Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 29, No. 7. pp. 3004–3019. DOI: 10.3906/elk-2101-123.
7. **Sánchez, P. A., Luna-Ramírez, S., Toril, M., Gijón, C., Bejarano-Luque, J. L. (2020).** A data-driven scheduler performance model for QoE assessment in a LTE radio network planning tool. Computer Networks, Vol. 173, 107186. DOI: 10.1016/j.comnet.2020.107186.
8. **Stafecka, A., Lizunovs, A., Ivanovs, G., Bobrovs, V. (2021)**. Dependence between signal parameter values and perceived internet access service QoS in mobile networks. 2021 Photonics & Electromagnetics Research Symposium (PIERS). pp 1419–1427. DOI: 10.1109/PIERS53385.2021.9695060.
9. International Telecommunication Union (2017) Quality of service regulation manual. In: Ind. Commun. Syst. https://www.itu.int/dms_pub/itu-d/opb/pref/D-PREF-BB.QOS_REG01-2017-PDF-E.pdf.
10. **Imoize, A. L., Orolu, K., Atayero, A. A. A. (2020).** Analysis of key performance indicators of a 4G LTE network based on experimental data obtained from a densely populated smart city. Data in Brief, Vol. 29, No. 105304. DOI: 10.1016/J.DIB.2020.105304.
11. **Zhang, L., Ni, Q., Zhai, M., Moreno, J., Briso, C. (2019)**. An ensemble learning scheme for indoor-outdoor classification based on KPIs of LTE network. IEEE Access, Vol. 7, pp. 63057–63065. DOI: 10.1109/ACCESS.2019.2914451.
12. **Tsolkas, D., Liotou, E., Passas, N., Merakos, L. (2017).** A survey on parametric QoE estimation for popular services. Journal of Network and Computer Applications, Vol. 77, pp. 1–17. DOI: 10.1016/j.jnca.2016.10.016.
13. **Stafecka, A., Lizunovs, A., Bobrovs, V. (2018).** Mobile LTE network signal and quality of service parameter evaluation from end-user premises. 2018 Advances in Wireless and Optical Communications, IEEE, pp. 209–212. DOI: 10.1109/RTUWO.2018.8587890.
14. **Türk, Y., Zeydan, E., Akbulut, C. A. (2018)**. An experimental analysis of differentiated quality of service support for LTE users. 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE), pp 408–412. DOI: 10.1109/ICEEE2.2018.8391372.
15. **Arteaga, C. P., Gomez, E. M., Florez, V. Q., Hernández, C. M. (2014).** A LTE basic system level simulator on capacity. 2014 IEEE Colombian Conference on Communications and Computing, IEEE, pp. 1–6. DOI: 10.1109/ColComCon.2014.6860423.
16. **Kurda, R. (2021).** Heterogeneous networks: Fair power allocation in LTE-A uplink scenarios. PloS One, Vol. 16, No. 6. DOI: 10.1371/journal.pone.0252421.

17. **Lai, L., Feng, D., Zheng, F. C., Wang, X., Yang, H. H., Quek, T. Q. (2021).** CQI-based interference detection and resource allocation with QoS provision in LTE-U systems. IEEE Transactions on Vehicular Technology, Vol. 70, No. 2, pp. 1421–1433. DOI: 10.1109/TVT.2021.3052530.
18. **Lai, W. K., Tang, C. L. (2013).** QoS-aware downlink packet scheduling for LTE networks. Computer Networks, Vol. 57, No. 7, pp. 1689–1698. DOI: 10.1016/j.comnet.2013.02.017.
19. **Bermudez, H. F., Martinez-Caro, J. M., Sanchez-Iborra, R., Arciniegas, J. L., Cano, M. D. (2019).** Live video-streaming evaluation using the ITU-T P.1203 QoE model in LTE networks. Computer Networks, Vol. 165, pp. 106967. DOI: 10.1016/J.COMNET.2019.106967.
20. **Torres-Vega, M., Mocanu, D. C., Stavrou, S., Liotta, A. (2017)**. Predictive no-reference assessment of video quality. Signal Processing: Image Communication, Vol. 52, pp. 20–32. DOI: 10.1016/j.image.2016.12.001.
21. **García-Pineda, M., Segura-García, J., Felici-Castell, S. (2018).** A holistic modeling for QoE estimation in live video streaming applications over LTE advanced technologies with full and non-reference approaches. Computer Communications, Vol. 117, pp. 13–23. DOI: 10.1016/j.comcom.2017.12.010.
22. **Vaser, M., Forconi, S. (2015).** QoS KPI and QoE KQI relationship for LTE video streaming and VoLTE services. 2015 9th International Conference on Next Generation Mobile Applications, Services and Technologies, IEEE, pp. 318–323. DOI: 10.1109/NGMAST.2015.34.
23. **Ghalut, T., Larijani, H., Shahrabi, A. (2016).** QoE-aware optimization of video stream downlink scheduling over LTE networks using RNNs and genetic algorithm. Procedia Computer Science, Vol. 94, pp. 232–239. DOI: 10.1016/J.PROCS.2016.08.036.
24. **Imoize, A. L., Tofade, S. O., Ughegbe, G. U., Anyasi, F. I., Isabona, J. (2022).** Updating analysis of key performance indicators of 4G LTE network with the prediction of missing values of critical network parameters based on experimental data from a dense urban environment. Data in Brief, Vol. 42, pp. 108240. DOI: 10.1016/j.dib.2022.108240.
25. **Sánchez, P. A., Luna-Ramirez, S., Toril, M. (2020).** A computationally efficient method for QoE-driven self-planning of antenna tilts in a LTE network. IEEE Access, Vol. 8, pp. 197005–197016. DOI: 10.1109/ACCESS.2020.3033325.
26. **Ferdosian, N., Othman, M., Ali, B. M., Lun, K. Y. (2015).** Throughput-aware resource allocation for QoS classes in LTE networks. Procedia Computer Science, Vol. 59, pp. 115–122. DOI: 10.1016/J.PROCS.2015.07.344.
27. **Uyan, O. G., Gungor, V. C. (2019).** QoS-aware LTE-A downlink scheduling algorithm: A case study on edge users. International Journal of Communication Systems, Vol. 32, No. 15, pp. e4066. DOI: 10.1002/DAC.4066.
28. **Zadeh, L. A. (1972)**. A rationale for fuzzy control. Journal of Dynamic Systems. Vol. 94, pp. 3–4. DOI: 10.1115/1.3426540.
29. **Tsipouras, M. G., Exarchos, T. P., Fotiadis, D. I. (2008).** A methodology for automated fuzzy model generation. Fuzzy Sets and Systems, Vol. 159, No. 23, pp. 3201–3220. DOI: 10.1016/J.FSS.2008.04.004.
30. **Zadeh, L. A., Aliev, R. A. (2018).** Fuzzy logic theory and applications: part I and part II. World Scientific Publishing.
31. **Sánchez-Fernández, L. P., Sánchez-Pérez, L. A., Concha-Gómez, P. D., Shaout, A. (2023).** Kinetic tremor analysis using wearable sensors and fuzzy inference systems in Parkinson's disease. Biomedical Signal Processing and Control, Vol. 84, p. 104748. DOI: 10.1016/J.BSPC.2023.104748.
32. **Ross, T. J. (2005).** Fuzzy logic with engineering applications. John Wiley & Sons.
33. **Jadhav, V. S., Kolekar, U. D. (2022).** Fuzzy-based decisive approach for call admission control in the LTE networks. Evolutionary Intelligence, Vol. 15, No. 2, pp. 1007-1024. DOI: 10.1007/S12065-019-00270-1.
34. **Liu, Z., Wang, Y., Xie, Y. A., Yuan, Y., Chan, K. Y., Yang, Y., Ma, K. (2021).** Interference-graph and fuzzy C-means based resource allocation scheme in LTE-V2V communication networks. AEU-International Journal of

Electronics and Communications, Vol. 142, p. 153982. DOI: 10.1016/J.AEUE.2021.153982.

35. **Khatib, E. J., Barco, R., Gómez-Andrades, A., Munoz, P., Serrano, I. (2015).** Data mining for fuzzy diagnosis systems in LTE networks. Expert Systems with Applications, Vol. 42, No. 21, pp. 7549–7559. DOI: 10.1016/J.ESWA.2015.05.031.
36. **Aboelezz, Z. A., Nafea, H. B., Zaki, F. W. (2019).** Handover and QoS control in LTE HetNet based on fuzzy logic design. 2019 7th International Japan-Africa Conference on Electronics, Communications, and Computations, (JAC-ECC), IEEE, pp. 178–183. DOI: 10.1109/JAC-ECC48896.2019.9051126.
37. **Nafea, H. B., Aboelezz, Z. A., Zaki, F. W. (2021).** Quality of service (QoS) for LTE network based on adaptive neuro fuzzy inference system. IET Communications, Vol. 15, No. 5, pp. 683–694. DOI: 10.1049/CMU2.12099.
38. **Sanchez-Fernández, L. P. (2021).** Environmental noise indicators and acoustic indexes based on fuzzy modelling for urban spaces. Ecological Indicators, Vol. 126, pp. 107631. DOI: 10.1016/j.ecolind.2021.107631.
39. **Hoßfeld, T., Wunderer, S., Beyer, A., Hall, A., Schwind, A., Gassner, C., Houidi, Z. B. (2020).** White paper on crowdsourced network and QoE measurements--definitions, use cases and challenges. DOI: 10.25972/OPUS-20232.
40. **ETSI (2023).** Welcome to the World of Standards! https://www.etsi.org/.
41. **BEREC (2014)**. Monitoring quality of Internet access services in the context of net neutrality. Report. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.berec.europa.eu/sites/default/files/files/document_register_store/2014/9/BoR_%2814%29_117_NN_QoS_Monitoring_Report.pdf.
42. **Ookla (2023).** https://www.ookla.com/.
43. **Recommendation ITU-T E.840 (2018)** Statistical framework for end-to-end network performance benchmark scoring and ranking. https://www.itu.int/rec/T-REC-E.840-201806-I/en.
44. **Ookla (2021).** Mexico's Mobile and Fixed Broadband Internet Speeds. https://www.speedtest.net/global-index/mexico#mobile.
45. **Official Gazette of the Federation (Mexico) (2021).** ACUERDO mediante el cual el Pleno del Instituto Federal de Telecomunicaciones establece los parámetros de banda ancha. DOF 27/12/2021. https://www.dof.gob.mx/nota_detalle.php?codigo=5639470&fecha=27/12/2021#gsc.tab=0.
46. **Jang, J. S. R., Sun, C. T., Mizutani, E. (1997).** Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE, Transactions on automatic control, Vol. 42, No. 10, pp. 1482–1484. DOI: 10.1109/TAC.1997.633847.
47. **Isabona, J., Imoize, A. L. (2021).** Terrain-based adaption of propagation model loss parameters using non-linear square regression. Journal of Engineering and Applied Science, 68, 1-19. DOI: 10.1186/s44147-021-00035-7.

Article received on 31/01/2024; accepted on 14/03/2024.
*Corresponding author is Luis Pastor Sánchez-Fernández.

ISSN 2007-9737

# Mobile Application with Voicebot for Detection of Family Violence: A Case Study in the District of Casa Grande, Perú

Jhoana Novoa-Gallardo[1], Morellia Rodriguez-Rodriguez[1], Javier Gamboa-Cruzado[1,2,*], Yesenia Vasquez Valencia[1], Jefferson López-Goycochea[3]

[1] Universidad César Vallejo,
Facultad de Ingeniería y Arquitectura, Lima,
Peru

[2] Universidad Nacional Mayor de San Marcos,
Facultad de Ingeniería de Sistemas e Informática, Lima,
Peru

[3] Universidad de San Martín de Porres,
Facultad de Ingeniería y Arquitectura,
Lima, Peru

jgamboa65@hotmail.com, {jnovoaga, mrodriguezro12}@ucvvirtual.edu.pe, yvasquez@ucv.edu.pe, jlopezg@usmp.pe

**Abstract.** The COVID-19 crisis, which emerged in early 2020 in Peru, has highlighted challenges in detecting and responding to domestic violence, with the District of Casa Grande being particularly affected. This urgent issue led to the need to develop more effective and faster tools to identify and address these cases. In this context, the presented research focused on the design and development of an innovative mobile application, equipped with a Voicebot, based on the Mobile-D methodology, known for its efficacy in mobile solutions development. To evaluate the utility and efficiency of this tool, a comprehensive and real case study was conducted using a pure experimental design, dividing the participants into two groups: an experimental one, which used the application, and a control group, which received no intervention. Four key indicators were measured: the time needed to detect a violent incident, the speed in connecting victims with nearby help centers, the total volume of cases identified, and the level of user satisfaction. The results obtained were encouraging: the application significantly reduced detection and response times, increased the number of detected cases, and significantly improved user satisfaction, proving to be an essential tool for addressing domestic violence in the District of Casa Grande.

**Keywords**. Mobile application, Voicebot, domestic violence detection, mobile-d methodology, satisfaction.

## 1 Introduction

Family violence is a global challenge, and the Casa Grande District has been no exception. In 2022, a significant increase in violent incidents was observed. The pandemic has intensified underlying situations, leading to economic strains, couple disputes, jealousy, and other triggering factors. Early detection of family violence is crucial in today's society.

In relation to this issue, the authors in [1] introduced an article titled “Mobile Phone Apps for Intimate Partner and Sexual Violence Prevention and Response: Systematic Search on App Stores.” Their main objective was to assess the quality of available mobile apps for the prevention and response to intimate partner and sexual violence.

Meanwhile, [2] in their research “Influence of Intrafamilial Violence on the Academic Performance of Adolescents,” examined the

repercussions of intrafamilial violence on the academic performance of adolescents from the Vicente Fierro educational unit in Tulcán. The findings revealed a significant impact of intrafamilial violence on the performance of students in 8th and 10th grades.

In their article "Health Care Counselling Via Voicebot Using Multinomial Naive Bayes Algorithm," the authors [3] proposed a Voicebot software designed to provide answers and solutions to users, as well as to assist in the diagnosis and treatment of patients, displaying symptoms, diagnoses, predictions, and prescriptions. In the study [4], titled "The use of mobile phone applications to enhance personal safety from interpersonal violence – an overview of available smartphone applications in the United Kingdom," 503 applications were reviewed, of which 83 met the evaluation criteria.

While many users found these applications helpful, concerns were also reported regarding the reliability and efficiency of some of them. As reported in [5], "Health Professionals' Experience Using an Azure Voice-Bot to Examine Cognitive Impairment (WAY2AGE)," the experience of 30 healthcare professionals using a VoiceBot to detect cognitive decline was analyzed. In [6], "Development and testing of an fpt.Ai-based voicebot," the authors introduced a voice robot that demonstrated an accuracy of 90.51% in open voice tests.

The author in [7], "Psychological Impact of COVID-19 on Children and Adolescents," addressed the psychological effects of Covid-19 on the youth, noting an increase in negative psychosocial factors during the pandemic. The article [8], "Violence towards Women and Coping Strategies in Family Mothers of Ucayali, Peru," explored the relationship between violence against women and their coping tactics, finding a significant inverse correlation.

In the research of [9], titled "Women who suffer from child maltreatment have a higher likelihood of being victims of partner violence in Peru," the relationship between child maltreatment and victimization by partner violence in Peruvian women over 18 years old was sought. The findings indicated that 49.6% of the women experienced child maltreatment, and 64.2% endured partner violence, highlighting the high vulnerability of women to violence.

On the other hand, the authors in [10], "Intimate Partner Violence in the Americas: A Systematic Review and Re-analysis of National Prevalence Estimates," emphasize that violence against women is a severe public health and human rights issue in the Americas, stressing the need for high-quality information to monitor and prevent violence. Likewise, the authors in [11], "Family Maltreatment as a Risk Factor for Antisocial Behavior in Adolescents," analyzed how the family environment can influence the antisocial behavior of adolescents, concluding that conduct disorder is multifactorial and does not differ between genders. Lastly, in the study [12], "Speech Recognition with Spanish Accent based on an Acoustic Model," an automatic voice recognition system is proposed, finding that the error rate varies depending on the number and duration of the audios, which can influence the model's accuracy.

Mobile applications aimed at detecting violence are essential in today's society, especially considering the increase in these incidents during the pandemic. These tools not only identify potential cases of domestic violence but also contribute to preventing future episodes. Each report entered into the application is meticulously registered by the Casa Grande Police Station.

This entity is responsible for delving into cases backed by evidence and looking for suitable solutions. Through this mobile application with Voicebot, not only are cases of violence addressed, but other requirements can also be channeled to the relevant authorities, ensuring a quick and efficient response.

This research aims to enhance the identification of domestic violence cases in the Casa Grande District through a mobile application with Voicebot, based on the Mobile-D methodology. Specific goals include: speeding up the detection process of violent incidents, accelerating alerts to nearby help centers, increasing the number of identified cases, and enhancing user satisfaction.

## 2 Theoretical Background

Mobile applications are software specifically designed to run on devices such as tablets and

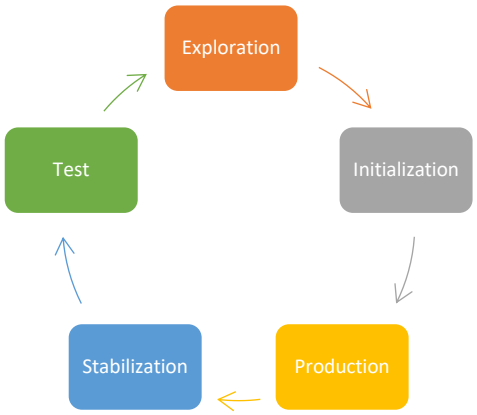



**Fig. 1.** Mobile-D methodology

**Table 1.** Operationalization of the dependent variable

| Indicator | Index | Unit of Measure | Unit of Observation |
|---|---|---|---|
| Time to detect family violence | [10 - 15] | minutes | Manual Review |
| Time to alert nearby help center | [5 - 10] | minutes | Manual Review |
| Number of cases detected | [0 - 6] | days | Direct observation |
| Level of Satisfaction | [Strongly Disagree - Strongly Agree] | Likert Scale | Observation Cards |

smartphones [13]. These programs have incorporated a broad range of functionalities that users of various devices leverage regularly as indicated by [14].

Among these applications, many offer services addressing situations of domestic violence [15]. On the other hand, voice recognition, as described in [16], is a technique that enables machines to decode and transcribe human speech using specific language models. Technology is advancing rapidly, and artificial intelligence, which seeks to emulate human thought, is a testament to this [17]. As services transition from physical to virtual, interactions between humans and technology intensify, leading to tools such as voicebots [18].

In the realm of mobile development, some of the most notable methodologies include Mobile-D, HMD, ASD, and NPD [19]. Regarding domestic violence, in [20] it is defined as harmful actions, be they verbal or physical, perpetrated by one family member towards another.

The study by Gamboa-Cruzado [43] reports that Mobile Applications allowed a significant reduction in access time, an increase in the number of workshop searches and an improvement in customer satisfaction. Finally, in [21] it is highlighted that electronic devices have evolved with the aim of enriching human life. Although their advanced use can be limiting for some, the positive impact of technological progress is undeniable.

## 3 Research Method

In this section, a detailed description of the methodological approach employed to conduct the research is provided. The procedures, techniques, and tools used are explained, along with the

rationale for their selection. Additionally, the experimental design, participant selection, data collection instruments, and analysis methods are presented.

This section is crucial for understanding the underlying methodology, how the results were obtained, and ensuring the replicability of the study.

### 3.1 Mobile-D Methodology

The Mobile-D Methodology was implemented following an agile approach for mobile application development in very small teams. According to the Mobile-D methodology, the goal is to obtain fully effective products in a period of less than ten weeks (See Fig. 1):

- Exploration: In this phase, project planning is initiated.
- Initialization: This second phase is dedicated to identifying the necessary requirements to start the project.
- Production: During the third phase, the technique known as Test Driven Development is employed to ensure higher quality in the project.
- Stabilization: In the fourth phase, integration actions are carried out to ensure that the system is complete and functioning correctly.
- System Testing and Adjustments: The objective of this phase is to achieve a stable version of the system that meets the client's needs and functions adequately.

### 3.2 Applied Research Methodology

In this subsection, the specific methodology applied in the study is delved into. Reasons behind the choice of this methodology, how it was adapted to the research context, and the concrete steps that were followed are discussed. Advantages and possible limitations of this approach will also be touched upon. Clarity in this section is crucial for readers to understand the validity and applicability of the results derived.

#### 3.2.1 Operationalization of Variables

In this section, the way each study variable has been operationalized is defined and detailed. This involves describing how the variables will be measured, categorized, or quantified in practical terms.

Operationalization is crucial to ensure the variables are measurable and that the study can be replicated (See Table 1).

#### 3.2.2 Research Design

The research design is pivotal in understanding how the study was conducted and how the variables were controlled to obtain valid and reliable results.

Research Design: Pure Experimental.

| | | |
|---|---|---|
| RGe | X | O1 |
| RGc | -- | O2 |

Data obtained from the experimental group (Ge), chosen through a random selection process (R), is based on the representative number of transactions related to the Detection of Family Violence and those who use the Mobile Application with Voicebot (X). Information from the control group (Gc) is also collected, to which the same stimulus is not administered (--).

#### 3.2.3 Universe and Sample

All processes for the detection of Family Violence at the National level were considered as the universe of the study. Hence, N = Indeterminate.

The sample is the process of detecting Family Violence in the district of Casa Grande. Where n=30.

#### 3.2.4 Universe and Sample

In the research, the data collection instrument was the Observation Form; and the techniques used were direct and indirect observation.

#### 3.2.5 Statement of Hypotheses

The following hypotheses were proposed: H1: If a Mobile Application with Voicebot is used, applying the Mobile-D Methodology, then the time to detect complaints is reduced.

H2: If a Mobile Application with Voicebot is used, applying the Mobile-D Methodology, then the search time for help centers is reduced.

**Table 2**. Functional requirements

| | Functional Requirements |
|---|---|
| RF 1 | The mobile application must display a login screen with the company logo with the company logo |
| RF 2 | The mobile application must allow the entry of ID and cell phone number data. |
| RF 3 | The mobile application must perform a quiz using VoiceBot. |
| RF 4 | The application must show a percentage of probability of family violence. |
| RF 5 | The mobile application must alert the police or help centers. |
| RF 6 | The application shall display user data. |
| RF 7 | The administrator shall have access to the app data through a web portal. |
| RF 8 | The web portal shall only be used by help centers. |

H3: If a Mobile Application with Voicebot is used, applying the Mobile-D Methodology, then the number of cases detected per day increases.

To contrast the hypotheses, the following solution was proposed for each of the indicators:

μ1 = Population mean (H1, H2, H3) for PostTest of Gc.

μ2 = Population mean (H1, H2, H3) for PostTest of Ge.

where: Ho: $\mu 1 < \mu 2$ and Ha: $\mu 1 \geq \mu 2$

Lastly, a test was carried out to determine data normality, a descriptive statistical analysis was performed (See Figure 13, Figure 14, and Figure 15), and to validate the hypotheses, the Student's t-test was used through the Minitab statistical software (See Table 9 and Table 10).

## 4 Case Study

In this section, a detailed case study is presented. The context, participants, gathered data, and observations made will be explored. The aim is to provide a deep understanding of a specific situation or problem, drawing from actual evidence and practical experiences.

Through this case study, the intention is to illustrate and validate the concepts and theories discussed earlier in the paper. The Mobile-D methodology was employed for the mobile application's development.

### 4.1 Exploration

In this phase, the initial process of identifying and understanding the project's requirements, needs, and expectations is detailed.

The activities conducted to gather information, the tools utilized, and the interactions with the stakeholders will be discussed. This phase is pivotal in establishing a solid foundation for the subsequent stages of development, ensuring the project aligns with the identified objectives and needs. Key findings, decisions made, and their justifications will be presented. (See Table 2, Table 3, and Table 4).

#### 4.1.1 Identifying Stakeholders

For the development of this activity, the following stakeholders were defined:

- Thesis author.
- Thesis advisor.

#### 4.1.2 Functional Requirements Definition

Table 2 details the functional requirements of the system under study, specifying necessary operations, inputs, and outputs.

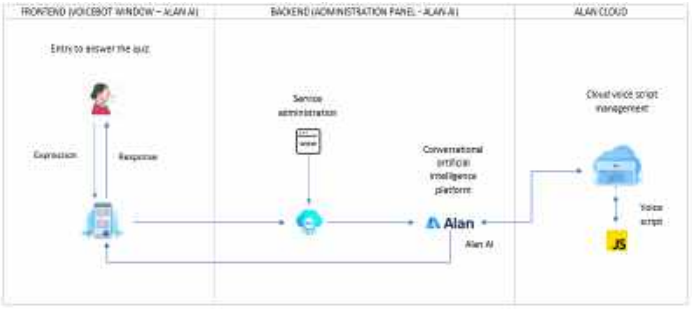



**Fig. 2.** Solution architecture

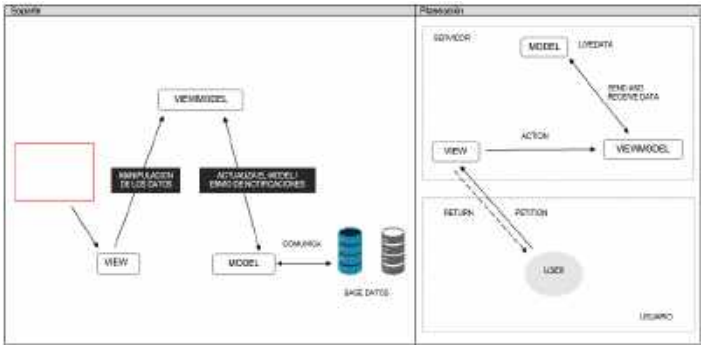



**Fig. 3.** Project architecture

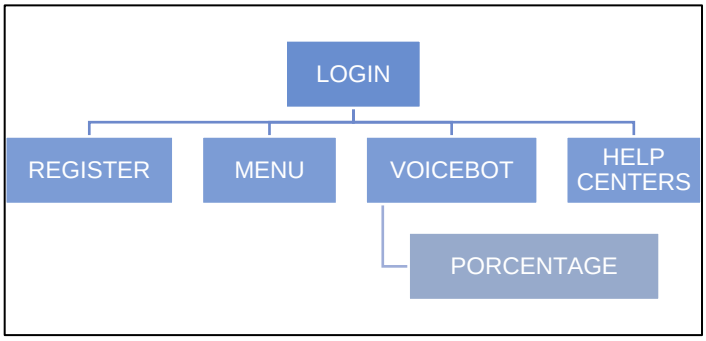



**Fig. 4.** Navigability scheme

### 4.1.3 Non-functional Requirements Definition

Non-functional requirements of the system are precisely identified and described. These cover aspects such as performance, security, usability, and compatibility, as shown in Table 3.

### 4.1.4 Development Tools

Table 4 provides a summary of the software and hardware tools used for the development of the system under study. Specifications of the programming environments, code libraries, version control systems, and testing platforms used are included.

## 4.2 Initialization

In this phase, the formal commencement of the project is described, laying down the foundation and necessary preparations for development. Initial configurations, role definitions, resource

**Table 3**. Non-functional requirements

| NON-functional Requirements | |
|---|---|
| RNF 1 | The application used the Mobile-D methodology |
| RNF 2 | The database will be stored in SQL Server. |
| RNF 3 | The mobile application was developed with Dart language, Php (Web Services). |
| RNF 4 | The mobile application must be easy to analyze and modify to correct failures. |

**Table 4.** Development tools

| Name | Description |
|---|---|
| Flutter | Flutter |
| Language | Dart |
| Visual Studio Code | Source code editor |
| PHP | Open source code |
| SQL Server | Relational database management system. |
| Hostinger | Extensible web server |

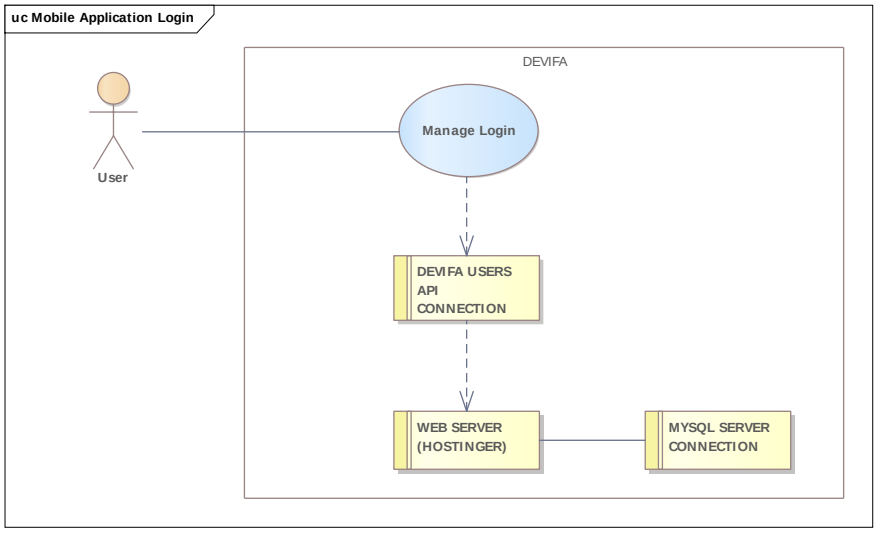



**Fig. 5.** Use case diagram: mobile application login

allocations, and preliminary planning are addressed.

This phase ensures that the project starts with a clear direction and all required elements in place. The solution's architecture, the project's architecture, and the navigability scheme were established. (See Fig. 2, Fig. 3, and Fig. 4).

For the project's creation, the MVC (Model-View-Controller) design pattern was used, which is divided into three components:

1 **Model**: Represents the data layer.
2 **View**: Represents the user interface.
3 **Controller**: Intermediate layer between data and interface.

ISSN 2007-9737



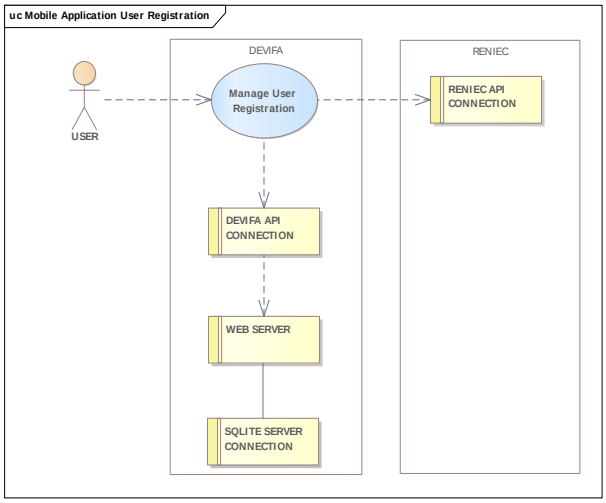



**Fig. 6.** Use case diagram: mobile application user registration

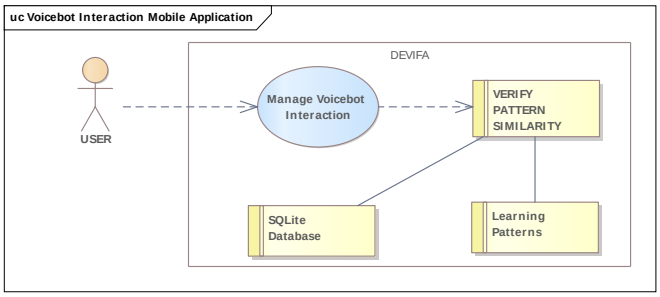



**Fig. 7.** Use case diagram: mobile application voicebot interaction

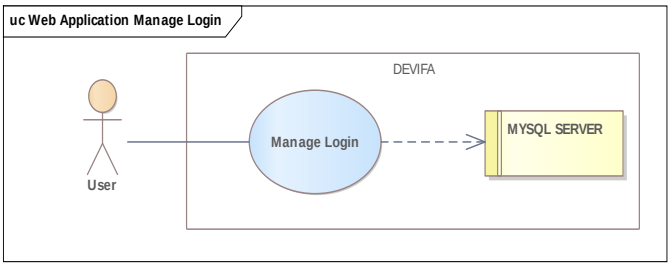



**Fig. 8.** Use case diagram: web application manage login

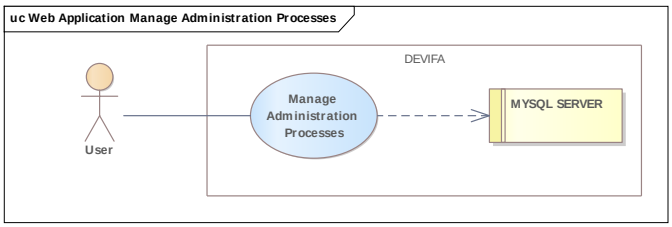



**Fig. 9.** Use case diagram: web application manage administration processes

**Table 5**. Dependency installation

| Dependency | Description |
|---|---|
| Dart | Programming language |
| Flutter | Development framework |
| Shared_preferences | Add-on for reading and writing simple key-value pairs |
| Http | Package for HTTP resource consumption (API) |
| provider | A wrapper around InheritedWidget to make them easier to use and more reusable make them easier to use and more reusable |
| url_launcher | Flutter plugin to launch a URL on Android and iOS |
| Progress_dialog | A lightweight package to display the progress dialog. |
| sqflite | Supports transactions and batches |
| Rxdart | It is an implementation of the popular reactiveX api for asynchronous programming. |
| Image selector | A Flutter plugin for iOS and Android to select images from the library. |
| Type_mine | It is a standardized way to indicate the nature and format of a document. |
| intl | It is the namespace for the ECMAScript Internationalization API. |
| Google_fonts | Allows you to easily use any of the 977 fonts (and their variants) from fonts.google.com |
| Time_picker_date | Provides a calendar as a horizontal timeline. |
| Cheer_up | An animation package inspired by Animate.css, built using only Flutter animations. |

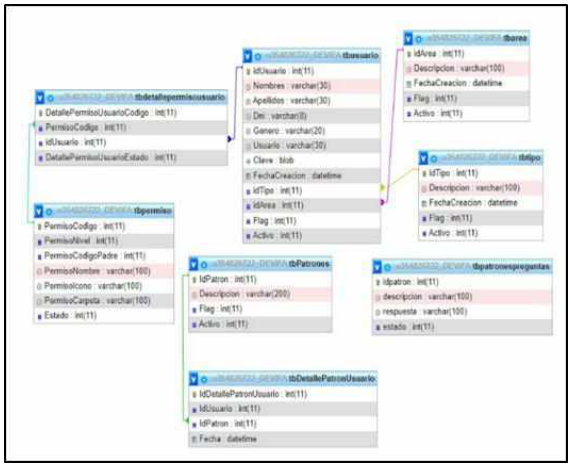


**Fig. 10.** Database diagram

### 4.3 Production

This phase is dedicated to describing the process where previous ideas and planning turn into tangible products. Activities related to coding, testing, integrations, and project refinements will be discussed.

The tools used, adopted work methodologies, and milestones reached will be highlighted.

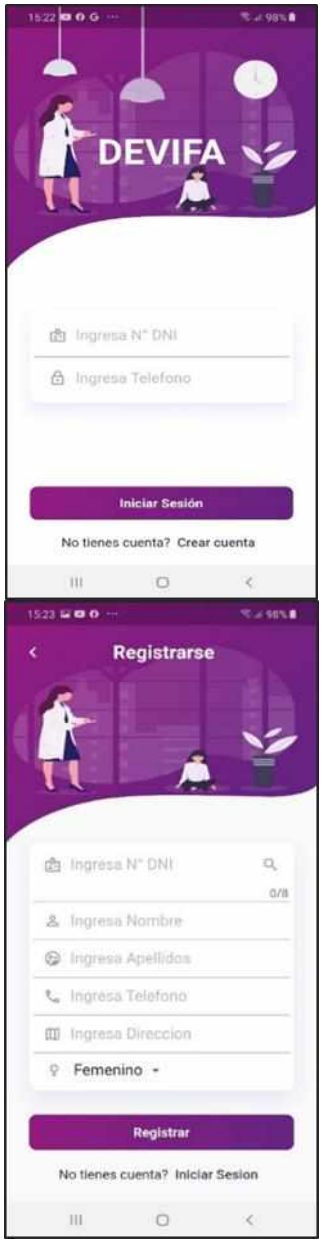



**Fig. 11.** Login and registration


**Fig. 12.** Questions and help center

Moreover, a detailed insight into how challenges were managed and how the final product's quality was ensured will be provided. (See Fig. 5, Fig. 6, Fig. 7, Fig. 8, and Fig. 9):

– Use Case Modeling.

### 4.4 Stabilization

In this phase, the process of refining and optimizing the project is addressed. Activities focused on identifying and correcting errors, enhancing performance, and ensuring the product

**Table 6**. Results for indicators

| | I1: Time to detect Family Violence | | I2: Time to alert nearby help centers | | I3: Number of cases detected | | I4: Level of Satisfaction (Likert scale) | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **PostTest Gc** | **PostTest Ge** | **PostTest Gc** | **PostTest Ge** | **PostTest Gc** | **PostTest Ge** | **PostTest Gc** | **PostTest Ge** |
| 1 | 1440 | 12,72 | 1080 | 4,51 | 3 | 4 | Strongly Disagree | Strongly Agree |
| 2 | 960 | 11,39 | 1140 | 3,74 | 0 | 3 | Strongly Disagree | Agree |
| 3 | 780 | 14,85 | 1320 | 3,89 | 1 | 3 | Neither Agree nor Disagree | Neither Agree nor Disagree |
| 4 | 1380 | 13,03 | 1380 | 4,76 | 3 | 5 | Strongly Disagree | Strongly Agree |
| 5 | 1440 | 14,68 | 1020 | 3,39 | 2 | 3 | Neither Agree nor Disagree | Strongly Agree |
| 6 | 1260 | 13,85 | 1380 | 4,46 | 2 | 6 | Neither Agree nor Disagree | Strongly Agree |
| 7 | 780 | 14,98 | 840 | 4,12 | 4 | 6 | Strongly Disagree | Agree |
| 8 | 1320 | 14,62 | 1320 | 4,05 | 0 | 2 | Strongly Disagree | Neither Agree nor Disagree |
| 9 | 780 | 12,10 | 1140 | 4,61 | 1 | 1 | Strongly Disagree | Agree |
| 10 | 1200 | 14,21 | 1440 | 4,24 | 2 | 3 | Strongly Disagree | Strongly Agree |
| 11 | 600 | 11,03 | 1200 | 4,99 | 1 | 6 | Neither Agree nor Disagree | Agree |
| 12 | 700 | 13,05 | 1080 | 4,19 | 4 | 5 | Strongly Disagree | Strongly Agree |
| 13 | 1080 | 13,41 | 840 | 3,08 | 3 | 5 | Neither Agree nor Disagree | Neither Agree nor Disagree |
| 14 | 1140 | 14,68 | 840 | 4,94 | 1 | 1 | Neither Agree nor Disagree | Strongly Agree |
| 15 | 900 | 12,44 | 780 | 3,90 | 6 | 6 | Strongly Disagree | Agree |
| 16 | 1320 | 11,49 | 720 | 4,36 | 3 | 5 | Strongly Disagree | Neither Agree nor Disagree |
| 17 | 780 | 14,48 | 600 | 4,45 | 4 | 0 | Strongly Disagree | Strongly Agree |
| 18 | 900 | 14,92 | 1020 | 3,02 | 0 | 2 | Strongly Disagree | Agree |
| 19 | 780 | 12,44 | 1140 | 4,18 | 2 | 4 | Neither Agree nor Disagree | Agree |
| 20 | 1140 | 12,95 | 1260 | 3,48 | 1 | 2 | Strongly Disagree | Strongly Agree |
| 21 | 600 | 13,85 | 1080 | 4,01 | 6 | 3 | Strongly Disagree | Strongly Agree |
| 22 | 1020 | 12,32 | 960 | 3,78 | 4 | 6 | Strongly Disagree | Agree |
| 23 | 1380 | 11,26 | 900 | 4,49 | 3 | 5 | Strongly Disagree | Neither Agree nor Disagree |
| 24 | 1140 | 12,79 | 780 | 3,29 | 0 | 4 | Neither Agree nor Disagree | Strongly Agree |
| 25 | 1140 | 13,48 | 600 | 3,10 | 3 | 4 | Strongly Disagree | Agree |
| 26 | 1380 | 13,21 | 1080 | 3,62 | 5 | 1 | Strongly Disagree | Strongly Agree |
| 27 | 1080 | 11,20 | 1200 | 4,20 | 0 | 4 | Strongly Disagree | Agree |
| 28 | 780 | 14,71 | 840 | 4,50 | 1 | 3 | Neither Agree nor Disagree | Strongly Agree |
| 29 | 1380 | 11,03 | 720 | 4,38 | 5 | 3 | Strongly Disagree | Strongly Agree |
| 30 | 900 | 14,04 | 1200 | 3,60 | 2 | 4 | Strongly Disagree | Strongly Agree |

meets the established quality standards will be tackled. It's a phase where the project's robustness is tested, and it is prepared for its final launch or implementation. Moreover, it provides a detailed insight into the efforts made to ensure the project is stable, reliable, and ready for delivery or deployment. (See Table 5 and Fig. 10):

– Dependency Installation.

### 4.5 Testing

In this phase, we detail the set of activities carried out to evaluate and validate the behavior of the developed product. The different types of tests conducted, whether unit, integration, system, or acceptance tests, among others, are addressed. This offers a deep understanding of how it was ensured that the project met the defined expectations and requirements and how it was prepared for its deployment and use in a real environment. (See Fig. 11 and Fig. 12):

- **Login**: Set of methods used to authenticate and enter the mobile application; one must log in using the DNI and cell phone number.
- **Registration**: For the Registration screen, it is mandatory to fill in fields such as: DNI, Phone, Address, and Gender.
- **Questions**: For the questions, there is a VoiceBot which will interact with the user and will ask a series of questions to detect Violence.
- **Help Centers**: On this screen, details of the community Help Centers will be displayed, which will be alerted once the user responds.

## 5 Results and Discussion

In this section, the results obtained from the experiments are presented and discussed in relation to existing literature and proposed hypotheses. It is vital to remember that the discussion is based on interpreting the results in light of theory and previous research.

### 5.1 Experimental Results

The results of this study clearly demonstrate that the Time to detect Family Violence, Time to alert nearby help centers, Number of Cases detected, and User Satisfaction Level have improved. 30 results were obtained for each indicator, as shown in Table 6.

### 5.2 Normality Test

Verifying normality is a fundamental prerequisite for many inferential statistical procedures that assume a normal distribution of residuals. In this study, the normality of data set distributions was assessed using various indicators and statistical tests. The Empirical Cumulative Distribution Function (ECDF) is a vital tool in this process. (See Fig. 13, Fig. 14, and Fig. 15).

**Indicator I1**: Time to detect Family Violence.

**Indicator I2**: Time to alert nearby help centers.

**Indicator I3**: Number of cases detected

For indicators I1, I2, and I3, the analysis reveals a p-value that lies below the set significance threshold ($\alpha = 0.05$). These results confirm that the data fit a normal distribution. Given this evidence, we will select the parametric Student's t-test to verify our hypotheses.

### 5.3 Discussion

This section plays a crucial role in the contextualization and interpretation of the findings obtained in the study. It aims to offer a deeper understanding and place the results within the framework of existing literature.

#### 5.3.1 Descriptive Statistics

An initial descriptive review of the data clearly shows several trends (See Table 7 and Table 8 for more details).

The results of the Anderson-Darling test indicate that, for most indicators and groups, the data is normally distributed. It is important to note that the indicator I1: PostTest Gc has a p-value of 0.067, which is close to the $\alpha$ (0.05) threshold. Although this value does not completely refute the normality of the data, it suggests that one should proceed with caution when interpreting the results for this indicator. The provided means and standard deviations reflect the central tendency and dispersion of the data, respectively, which is consistent with our observations about the normality of the distribution

Three key indicators, I1, I2, and I3, are presented, along with their respective 95% confidence intervals. For indicator I1, the average post-test time ranges from 12,504 to 13,997 minutes, and the third quartile (Q3) indicates that 75% of observations are below 14,515 minutes.

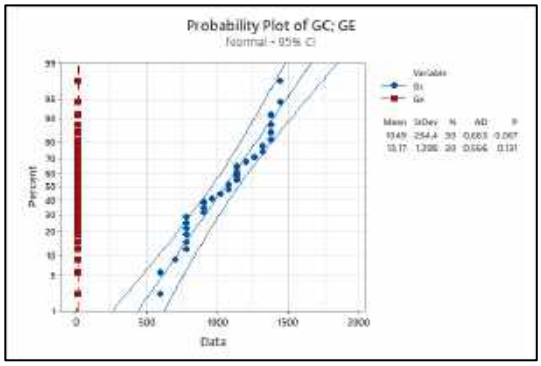



**Fig. 13.** Time to detect family violence

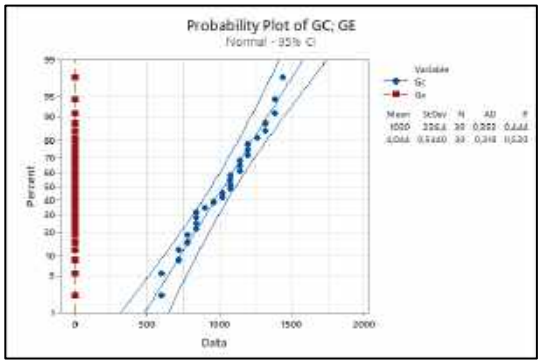



**Fig. 14.** Time to alert help centers

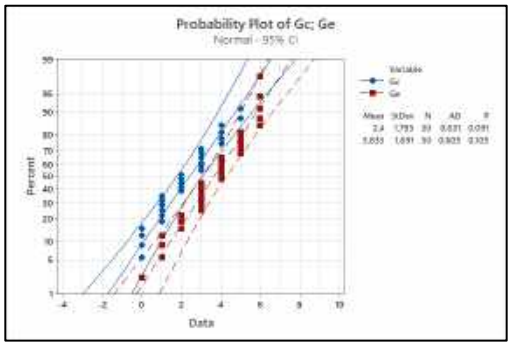



**Fig. 15.** Number of cases detected

Indicator I2 shows a range from 3,8052 to 4,3754 minutes with a Q3 of 4,4675 minutes.

For indicator I3, between 3,0000 and 4,7713 daily detected cases are reported, and a Q3 suggests that 75% of the records report fewer than 5 cases daily. Notably, the kurtosis values for I1, I2, and I3 are negative, suggesting tails lighter than a normal distribution.

Regarding skewness, all the indicators display negative skewness, indicating a greater concentration of data towards the higher values.

**For Indicator 1**, the results obtained align with those of [22], which detected a significant 79.3% reduction in the time needed to identify a reported case of domestic violence. Similarly, [23] documented a 72% reduction in domestic violence

**Table 7**. Results with descriptive statistics

| Indicator | n | Mean | StDev | AD | p-value |
|---|---|---|---|---|---|
| I1: PostTest Gc | 30 | 1049 | 264,4 | 0,683 | 0,067 |
| I1: PostTest Ge | | 13,17 | 1,298 | 0,566 | 0,131 |
| I2: PostTest Gc | 30 | 1030 | 236,4 | 0,352 | 0,444 |
| I2: PostTest Ge | | 4,044 | 0,5440 | 0,318 | 0,520 |
| I3: PostTest Gc | 30 | 2,4 | 1,793 | 0,631 | 0,091 |
| I3: PostTest Ge | | 3,633 | 1,691 | 0,605 | 0,105 |

**Table 8**. Summary of results for the Indicators

| Indicator | n | 95% Confidence Intervals for the Mean | Kurtosis | Skewness | Q3 |
|---|---|---|---|---|---|
| I1: PostTest Ge | 30 | 12,504 – 13,997 min | -1,17701 | -0,22951 min | 14,515 min |
| I2: PostTest Ge | 30 | 3,8052-4,3754 min | -0,684994 | -0,298199 min | 4,4675 min |
| I3: PostTest Ge | 30 | 3,0000 – 4, 7713 cases | -0.666061 | -0.292097 | 5,0000 |

detection time. In agreement, the authors in [25] confirm this trend, indicating a 32.9% decrease. However, these findings exceed those of [26], where only a 20% reduction in detection time was observed.

Given this evidence and the use of Mobile – D in the current research, the substantial reduction in the time it takes to detect domestic violence is emphasized, suggesting the potential for replicating these results in various contexts.

**In relation to Indicator 2**, the data gathered resemble those of [27], showing a 33.39% reduction in the time to alert assistance centers. Along the same lines, [28] reflected a 49% decrease, and [29] a 36% decrease. Additionally, [30] and [31] reported a reduction of 63.2% and 32% in the time required to send alerts to assistance centers, respectively.

As a result, based on this study utilizing Mobile – D, it's inferred that we have achieved tangible accomplishments, evidenced by the reduced alert time to assistance centers. It is crucial to consider all participants in the process.

**Regarding Indicator 3**, when compared with [32], they reported an increase of 5,937 detected domestic violence cases. Analogously, [33] recorded a significant 42% rise in the number of detected violence cases, regardless of their type. Continuing this trend, [34] highlights a specific vulnerability in children, with a daily increase of 47% in reports. Coherently, [35] emphasizes a 12.5% growth in incidents related to domestic violence. Meanwhile, [38] points to a 53.2% surge in the total reported domestic violence situations.

In this light, it's inferred that the study conducted using Mobile – D has yielded favorable

**Table 9**. Hypothesis testing for parametric indicators

| Sample | n | $H_0$ | t-value | p-value |
|---|---|---|---|---|
| I1: PostTest(Gc) / I1: PostTest(Ge) | 30 | μ1 ≤ μ2 | 21.46 | 0.000 |
| I2: PostTest(Gc) / I2: PostTest(Ge) | 30 | μ1 ≤ μ2 | 23.77 | 0.000 |
| I3: PostTest(Gc) / I3: PostTest(Ge) | 30 | μ1 ≥ μ2 | -2,74 | 0.004 |

**Table 10**. Hypothesis testing for non-parametric indicators

| Sample | n | $H_0$ | w-value | p-value |
|---|---|---|---|---|
| I4: PostTest (Gc) / I4: PostTest (Ge) | 30 | μ1 ≤ μ2 | 487.50 | 0.000 |

findings, showcasing an increase in identified cases. It's imperative to address these results, giving special consideration to the victims.

#### 5.3.2 Inferential Statistics (Hypothesis Testing)

Table 9 and Table 10 display the findings of the statistical analyses conducted to assess the proposed hypotheses.

Given that all the obtained p-values are less than α (0.05), the results provide compelling evidence to reject the null hypotheses (Ho) and thus give credence to the alternative hypotheses. This implies that the variables analyzed have a significant relationship and that the differences or associations found are not merely due to chance.

In this context, it's crucial to highlight the significance of these findings, as they indicate that the tests conducted are statistically significant, bolstering the validity and potential impact of the conclusions drawn from the research.

## 6 Conclusions and Future Research

The mobile application developed based on the Mobile-D Methodology has significantly optimized the process of identifying Family Violence cases in the Casa Grande district. We evaluated our main variable through four indicators: time to identify family violence, time to notify nearby help centers, the number of cases identified daily, and user satisfaction level.

After the software's implementation, there was a reduction in the time required to detect family violence, a decrease in the period needed to alert nearby help centers, an increase in the number of cases identified daily, and an improvement in user satisfaction.

For future research, it is suggested to continue using the Mobile-D methodology, given its effectiveness and thoroughness in developing mobile applications of various kinds. It is also advisable for the software to adapt to address more vulnerable cases and to expand its application on provincial, regional, or national scales.

## References

1. **Moret, J. D., Todd, A., Rose, L., Pollitt, E., Anderson, J. (2022).** Mobile phone apps for intimate partner and sexual violence prevention and response: systematic search

on app stores. JMIR formative research, Vol. 6, No. 2, p. e28959. DOI: 10.2196/28959.

2. **Morillo--Cano, J. R., Guerrón-Enriquez, S. X., Narváez-Jaramillo, M. E. (2021).** Influencia de la violencia intrafamiliar en el rendimiento académico de adolescentes. Conrado, Vol. 17, No. 81, pp. 330–337.
3. **Revathy, S., Niranjani, R., Roslin-Kanushya, J. (2020)**. Health care counselling via voicebot using multinomial Naive Bayes algorithm. Proceedings of 5th International Conference on Communication and Electronics Systems, pp. 1063–1067. DOI: 10.1109/ICCES48766.2020.9137948.
4. **Ford, K., Bellis, M. A., Judd, N., Griffith, N., Hughes, K. (2022).** The use of mobile phone applications to enhance personal safety from interpersonal violence–an overview of available smartphone applications in the United Kingdom. BMC public health, Vol. 22, No. 1, p. 1158. DOI: 10.1186/s12889-022-13551-9.
5. **Moret-Tatay, C., Radawski, H. M., Guariglia, C. (2022).** Health professionals' experience using an azure voice-bot to examine cognitive impairment (WAY2AGE). Healthcare, Vol. 10, No. 5, p. 783. DOI: 10.3390/healthcare 10050783.
6. **Tran, D. C., Nguyen, D. L., Hassan, M. F. (2020).** Development and testing of an FPT. AI-based voicebot. Bulletin of Electrical Engineering and Informatics, Vol. 9, No. 6, pp. 2388–2395.
7. **Sánchez-Boris, I. M. (2021).** Impacto psicológico de la COVID-19 en niños y adolescentes. Medisan, Vol. 25, No. 1, pp. 123–141.
8. **Carrión-Abarca, F., Turpo. J. A. (2022)**. Violencia hacia la mujer y estrategias de afrontamiento en madres de familia de Ucayali. Perú, Apuntes Universitarios Vol. 12, No. 3.
9. **Rodríguez-de-la-Cruz, M. C., Alarco, J. J. (2021)**. Las mujeres que sufren de maltrato infantil tienen una mayor probabilidad de ser víctimas de violencia ejercida por la pareja en Perú. Revista Brasileira de Epidemiologia, Vol. 24, p. e210058.
10. **Bott, S., Guedes, A., Ruiz-Celis, A. P., Mendoza, J. A. (2021).** La violencia por parte de la pareja íntima en las Américas: una revisión sistemática y reanálisis de las estimaciones nacionales de prevalencia. Revista Panamericana de Salud Pública, Vol. 45, p. e34. DOI: 10.26633/rpsp.2021.34.
11. **Vera-Sánchez, L. J., Alay-Giler, A. (2021).** El maltrato en la familia como factor de riesgo de conducta antisocial en adolescentes. Revista de Ciencias Humanísticas y Sociales (ReHuSo), Vol. 6, No. 1, pp. 23–40. DOI: 10. 5281/zenodo.5512717.
12. **Acosta-Urigüen, M. I., Cedillo-Orellana, I. P., Plaza-Salto, J. G., Sánchez-Zhunio, C., Zambrano-Martinez, J. L., Orellana-Cordero, M. P.** Reconocimiento del habla con acento español basado en un modelo acústico. Enfoque UTE, Vol. 13, No. 3, pp. 45–57. DOI: 10.29019/enfoqueute.839.
13. **Ortega-Martín, M. E., Lucena-Antón, D., Luque-Moreno, C., Heredia-Rizo, A. M., Moral-Munoz, J. A. (2020).** Aplicaciones móviles en el abordaje terapéutico del ictus: Revisión en repositorios comerciales y búsqueda de evidencia. Revista Española de Salud Pública, Vol. 93, pe201906035.
14. **Molina-Ríos, J. R., Honores-Tapia, J. A., Pedreira-Souto, N., Pardo, H. (2021)**. Estado del arte: metodologías de desarrollo de aplicaciones móviles. 3C Tecnología, Vol. 10, No. 2, pp. 17–45. DOI: 10.17993/3ctecno/ 2021.v10n2e38.17-45.
15. **Prado, C., Netto, A. V., Berton, L., Takahara, A. K. (2021).** Aplicação de healthbots em língua portuguesa: revisão narrativa. Journal of Health Informatics, Vol. 13, No. 4.
16. **Gottardi, W., de-Almeida, J. F. Soufen-Tumolo, C. H. (2022)**. Automatic speech recognition and text-to-speech technologies for L2 pronunciation improvement: reflections on their affordances. Texto Livre 15. DOI: 10.35699/1983-3652.2022.36736.
17. **Porcelli, A. M. (2020)**. La Inteligencia artificial y la robótica: sus dilemas sociales, éticos y jurídicos. Derecho Global, Estudios Sobre Derecho y Justicia, Vol. 6, No. 16, pp. 49–105. DOI: 10.32870/dgedj.v6i16.286.

18. **Russell-Bennett, R., Scott-Rosenbaum, M. (2022)**. Editorial: Opportunities in the new service marketplace. Journal of Service Marketing, DOI: 10.1108/JSM-04-2022-0121.
19. **Molina-Ríos, J. R., Honores-Tapia, J. A., Pedreira-Souto, N., Pardo-León, H. (2021).** Comparativa de metodologías de desarrollo de aplicaciones móviles. 3c Tecnología: glosas de innovación aplicadas a la pyme, Vol. 10, No. 2, pp. 73–93. DOI: 10.17993/3ctecno/2021.v10n2e38.73-93.
20. **Mas-Camacho, M. R., Acebo-del-Valle, G. M., Gaibor-González, M. I., Chávez-Chacán, P. J., Núñez-Aguiar, F. R., González-Nájera, L. M., Guarnizo-Delgado, J. B., Gruezo-González, C. A. (2020).** Violencia intrafamiliar y su repercusión en menores de la provincia de Bolívar, Ecuador. Revista Colombiana de Psiquiatría, Vol. 49, No. 1, pp. 23–28. DOI: 10.1016/j.rcp.2018.04.006.
21. **Chandini, A., Bhaskar-Reddy. P. V. (2020)**. Smart home automation using a voice-bot. International Journal of Advanced Research in Computer Science, Vol. 11. DOI: 10.26483/ijarcs.v11i0.6584.
22. **Sologuren, N. S. (2019)**. Protección jurídica de la mujer que denuncia violencia en el ámbito familiar. La Vida & La Historia, No. 9, pp. 41–50. DOI: 10.33326/26176041.2019.9.792.
23. **Benavides-Román, A. M., Rivera-Fernández, D. H., Chávez-Yomona. M. (2022)**. Violencia familiar y trastorno depresivo en mujeres e integrantes del grupo familiar en la jurisdicción de la Comisaría de familia de Lima. Revista de Derecho, Vol. 7, No. 1. DOI: 10.47712/rd.2022.v7i1.163.
24. **Jurado-Flores, V. D., Fuentes-Ríos, F. J. (2022).** Concentración espacial de las denuncias por violencia familiar en Ciudad Victoria: un acercamiento mediante la detección de hot spots. CienciaUAT, Vol. 16, No. 2, pp. 126–140. DOI: 10.29059/cienciauat.v16i2.1551.
25. **Dávalos-Martínez, A., Barrera-Rodríguez, E., Emigdio-Vargas, A., Blanco-García, N. O., Vélez-Núñez, B. (2021).** Funcionalidad familiar y violencia en mujeres adolescentes de Acapulco, México. Dilemas contemporáneos: educación, política y valores, (SPE4), Vol. 8.
26. **Gonzáles-Alvarenga, R. C. (2020)**. Violencia familiar: Estudio de casos de la unidad especializada de la fiscalía regional de Ciudad del Este. Revista jurídica, Investigación en Ciencias jurídicas y Sociales, Vol. 2, No. 10, pp. 137–51.
27. **Cevallos-Altamirano, A. S. (2021)**. Incidencia de la violencia contra la mujer y miembros del núcleo familiar en tiempos de pandemia. Revista Jurídica Crítica y Derecho, Vol. 2, No. 3, pp. 11–29.
28. **Herrera-Hugo, B. Á., Cárdenas-Lata, B. J., Tapia-Segarra, J. I., Calderón-Bustamante, K. N. (2021)**. Violencia intrafamiliar en tiempos de Covid-19: Una mirada actual. Polo del Conocimiento, Vol. 6, No. 2, p. 1027. DOI: 10.23857/pc.v6i2.2334.
29. **Jandres, M., Hernández, C. E., Castro, M. (2019).** Conocimientos, barreras y facilitadores para atención de violencia en área metropolitana de El Salvador, 2018. Alerta, Revista científica del Instituto Nacional de Salud, Vol. 2, No. 1, pp. 51–59. DOI: 10.5377/alerta.v2i1.7527.
30. **Murga-Alayo, L. E., Bermudez-Bailon, D. E. (2022)**. Medidas de protección emitidas en tiempo de pandemia para controlar la violencia física y psicológica. Revista Científica WARMI Intervención en Violencia Contra Las Mujeres, Vol. 2, No. 1. DOI: 10.46363/warmi.v2i1.4.
31. **Rojas-Gonzalez E. P. (2021)**. Violencia de género ante el confinamiento e intervención policial por causa de la covid-19 en Ciudad Juárez, Chihuahua. Revista Iberoamericana de las Ciencias Sociales y Humanísticas, Vol. 10, No. 19. DOI: 10.23913/ricsh.v10i19.227.
32. **Orozco-Aguancha, K., Jiménez-Ruiz, L. K., Cudris-Torres, L. (2020)**. Mujeres víctimas de violencia intrafamiliar en el norte de Colombia. Revista de Ciencias Sociales, Vol. 26, No. 2.
33. **Caqui-Pajuelo, Y. M. (2020)**. Violencia familiar contra la mujer: análisis desde un enfoque personalista. Apuntes de Bioética,

Vol. 3, No. 2, pp. 62–80. DOI: 10.35383/apuntes.v3i2.494.

34. **Abasolo-Telleria, A. E. (2019)**. Estudio descriptivo del tipo de maltrato que sufren menores evaluados en la Unidad de valoración forense integral de Bizkaia. Revista española de medicina legal: órgano de la Asociación Nacional de Médicos Forenses. Vol. 45, No. 1, pp. 4–11. DOI: 10.1016/j.reml.2018.04.004.
35. **Reingle-Gonzalez, J. M., Molsberry, R., Maskaly, J., Jetelina, K. K. (2020).** Trends in family violence are not causally associated with COVID-19 stay-at-home orders: A commentary on Piquero et al. American Journal of Criminal Justice, Vol. 45, No. 6, pp. 1100–1110. DOI: 10.1007/s12103-020-09574-w.
36. **Rivera, R., Arias-Gallegos, W. L. (2020)**. Factores asociados a la violencia contra los adolescentes dentro del hogar en el Perú. Interacciones, Vol. 6, No. 3. DOI: 10.24016/2020.v6n3.104.
37. **Villavicencio-Cedeño, J. A., Flores-Marín, M. (2018).** Efectos de la utilización de la aplicación móvil VILLAHEALTH en el comportamiento y desempeño del personal de enfermería. Revista Cubana de Enfermería, Vol. 34, No. 2.
38. **Eisenhut, K., Sauerborn, E., García-Moreno, C., Wild, V. (2020)**. Mobile applications addressing violence against women: a systematic review. BMJ Global Health, Vol. 5, No. 4, p. e001954, DOI: 10.1136/bmjgh-2019-001954.
39. **Morillo, P., Navarro-Pérez, J. J., Orduña, J. M., Fernández, M. (2022).** Evaluation of an intervention program based on mobile apps to learn sexism prevention in teenagers. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), Vol. 18, No. 2, pp. 1–20**.** DOI: 10.1145/3471139.
40. **Paniagua, L. A., Bedoya, R. D., Mera, C. (2020).** A method for assessing the accessibility and usability in mobile applications. TecnoLógicas, Vol. 23, No. 48, pp. 98–116. DOI: /10.22430/22565337.1553.
41. **De-la-Rosa-Gómez, A., Miranda-Díaz, G. Mendoza, S. (2020)**. Usabilidad y satisfacción de una aplicación móvil para el entrenamiento de competencias clínicas. Hamut'ay, Vol. 7, No. 1, pp. 48–59. DOI: 10.21503/hamu.v7i1.1908.
42. **Moret-Tatay, C., Radawski, H. M., Guariglia, C. (2022).** Health Professionals' experience using an azure voice-bot to examine cognitive impairment (WAY2AGE). Healthcare, Vol. 10, No. 5, p. 783. DOI: 10.3390/healthcare10050783.
43. **Chiclayo-Silvestre, A., Nizama-Florian, J., Gamboa-Cruzado, J., Vásquez-Valencia, Y., Tello-Aguilar, C. (2024).** Maskay-aplicación móvil para la búsqueda y acceso a talleres mecánicos: Un estudio de caso en Trujillo, Perú. Revista Ibérica de Sistemas e Tecnologias de Informação, (E65), pp. 204–219.

*Article received on 01/11/2023; accepted on 17/03/2024.*
**Corresponding author is J Javier Gamboa-Cruzado.*

ISSN 2007-9737

# Deflexiones en trabes de sección transversal rectangular con cartelas parabólicas sometidas a una carga concentrada

Rogelio Barraza-Saucedo, Arnulfo Luévanos-Rojas*

Universidad Autónoma de Coahuila,
Instituto de Investigaciones Multidisciplinaria,
México

rogelio.barraza.s@gmail.com, arnulfol_2007@hotmail.com

**Resumen.** Este documento presenta un modelo analítico para obtener las rotaciones y los desplazamientos perpendiculares al eje longitudinal en cualquier punto de una viga de sección rectangular con cartelas parabólicas sometida a un momento en cada apoyo y una carga concentrada tomando en cuenta las deformaciones por flexión y cortante, que es la principal contribución de esta investigación. El modelo actual no considera las deformaciones por cortante para obtener las rotaciones y los desplazamientos perpendiculares al eje longitudinal de la viga. La metodología se realiza bajo el concepto de que la derivada del desplazamiento vertical es la pendiente o rotación y la segunda derivada es el momento de la viga. También, se realiza una comparación entre el modelo propuesto y el modelo actual para vigas simplemente apoyadas, vigas empotradas en ambos extremos y vigas usando los mismos momentos y reacciones en los extremos (modelo propuesto) para ambos modelos con respecto al desplazamiento vertical máximo y su ubicación en la viga. Los resultados muestran que el modelo propuesto es mayor para vigas simplemente apoyadas y vigas usando los mismos momentos y reacciones en los extremos, y menor para vigas empotradas en ambos extremos para el desplazamiento vertical máximo con respecto al modelo actual. Por tanto, el modelo propuesto es más seguro y más adecuado con respecto al modelo actual para el análisis estructural, ya que las fuerzas cortantes y los momentos están presentes en todas las estructuras y se producen deformaciones por flexión y cortante.

**Palabras clave.** Trabes rectangulares, cartelas parabólicas, deformaciones por flexión y cortante, rotaciones, desplazamientos verticales.

## Deflections in Beams of Rectangular Cross Section with Parabolic Haunches Subjected to a Concentrated Load

**Abstract.** This paper introduces an analytical model to determine the rotations and displacements perpendicular to the longitudinal axis at any point on a rectangular-section beam with parabolic stiffeners, exposed to moments at each support and a concentrated load, considering both bending and shear deformations, which is the primary contribution of this research. The current model does not account for shear deformations when calculating the rotations and displacements perpendicular to the beam's longitudinal axis. The methodology relies on the vertical displacement's first derivative as the slope or rotation and the second as the moment. Additionally, the proposed model is compared to the current model for simply supported beams, beams fixed at both ends, and beams with identical moments and reactions at the ends. The comparison focuses on the maximum vertical displacement and its location along the beam. The results show that the proposed model yields greater values for simply supported beams and beams using the same moments and reactions at the ends, and smaller values for beams fixed at both ends, in terms of the maximum vertical displacement compared to the current model. Therefore, the proposed model is safer and suitable than the current one for structural analysis, as shear forces and moments are present in all structures, and both bending and shear deformations occur.

**Keywords.** Rectangular beams, parabolic haunches, bending and shear deformations, rotations, vertical displacements.

ISSN 2007-9737



## 1. Introducción

La deflexión de las vigas es el movimiento hacia abajo que hace una viga desde su posición inicial sin carga a otra posición deformada cuando se le aplica una carga. Dado que las vigas suelen tratarse como elementos bidimensionales, el eje neutro de una viga suele tomarse como punto de referencia para medir la deflexión de las vigas.

La deformación de vigas de acero estructural y concreto reforzado es una medida importante de su rendimiento de servicio, porque se requiere cumplir con los códigos de diseño actuales, y las deformaciones en las vigas consisten en deformaciones por flexión y cortante.

Muchos investigadores han estudiado la curva elástica de trabes para obtener las deflexiones máximas, pero no consideran las deformaciones por cortante. Algunos investigadores han presentado estudios sobre vigas en Voladizo no prismáticas, tales como:

Lee [9], Dado y Al-Sadder [6], Borboni y De Santis [2], Banerjee et al. [1], Solano-Carrillo [21], Chen [4], Yau [25], Brojan et al. [3].

Algunos investigadores han desarrollado modelos matemáticos para obtener la curva elástica para vigas simplemente apoyadas de sección rectangular constante sometidas a una carga uniformemente distribuida y/o carga concentrada tomando en cuenta las deformaciones por flexión y cortante para obtener las deflexiones en cualquier lugar de la viga (rotaciones y desplazamientos perpendiculares al eje longitudinal) [14, 16].

Otros autores han desarrollado modelos matemáticos y ayudas de diseño para vigas rectangulares con cartelas rectas o cartelas parabólicas en los extremos para las vigas sometidas a una carga uniformemente distribuida o carga concentrada tomando en cuenta las deformaciones a flexión y cortante [12, 13, 24, 19, 18, 8, 5, 11].

Estos trabajos muestran los factores para los momentos de empotramiento, factores de transporte o factores de arrastre y factores de rigidez. Varios autores han presentado modelos óptimos para obtener los lados de la viga y su acero de refuerzo longitudinal para secciones rectangulares con cartelas rectas o cartelas parabólicas bajo carga uniformemente distribuida o carga concentrada y momentos en los extremos [15, 10]. Las revisiones de la literatura de las investigaciones desarrolladas y/o comparadas por software se muestran a continuación: Majumder y Kumar [17] estudiaron la deflexión máxima de una viga simplemente apoyada sometida a tres tipos de carga. Las cargas son:

a) Carga uniformemente distribuida.
b) Carga concentrada en el centro de la viga.
c) Carga triangularmente distribuida.

El análisis teórico se realizó por la Teoría de Euler-Bernoulli (Deformaciones por flexión) y se comparó con el software ANSYS 14.0.

Los resultados numéricos mostraron una excelente precisión entre la teoría de Euler-Bernoulli y el software ANSYS 14.0. Debnath y Debnath [7] analizaron la deflexión máxima para diferentes tipos de vigas de sección transversal rectangular constante:

1. Vigas simplemente apoyadas:
   a) Carga uniformemente distribuida,
   b) Carga concentrada en el centro de la viga.
2. Vigas en Voladizo:
   a) Carga uniformemente distribuida,
   b) Carga concentrada en el extremo libre.

El cálculo teórico se realizó según la Teoría de Euler-Bernoulli (Deformaciones por flexión) y se comparó con el software ANSYS 14.0. Los datos considerados para las vigas son:

- $L = 100\,\text{m}$,
- $b = 10\,\text{m}$,
- $h = 10\,\text{m}$,
- $\nu = 0.3$,
- $E = 2 \times 10^7\ \text{N/m}^2$,
- $F = 500\,\text{N}$.

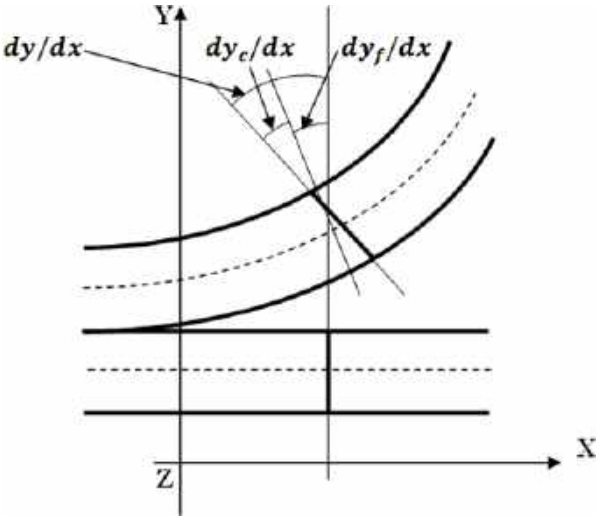



**Fig. 1.** Deformación de un elemento de viga

Los elementos sólidos estudiados fueron 188, 189, 185 y 285. El resultado más preciso fue medido por el elemento sólido 189 seguido por el elemento sólido 188 y otros elementos sólidos. los resultados fueron precisos entre la teoría de Euler-Bernoulli y el software ANSYS 14.0.

Sihua et al. [20] presentaron el análisis no lineal para una viga de concreto reforzado con ayuda del software de análisis de elementos finitos ABAQUS. La viga es simplemente apoyada y tiene $1500\,\text{mm}$ de largo; con una sección de $180 \times 1000$ mm, la resistencia del concreto es C25, el refuerzo longitudinal y estribos adoptados HPB235 reforzados.

En este análisis de vigas simplemente apoyadas, se ha introducido a fondo el modelo de plasticidad del daño del concreto en ABAQUS. Finalmente, los resultados de la experimentación y el análisis ABAQUS se compararon en un diagrama, y la carga alcanzó la capacidad de $24\,\text{kN}$, el valor de deflexión en el centro de la longitud de la viga es de $10.521\,\text{mm}$ de ABAQUS y $12.795\,\text{mm}$ de prueba.

Si se hubieran considerado las deformaciones de cortante en el software ABAQUS, los resultados del software ABAQUS estarían más cerca de la prueba experimental, porque estas deformaciones tienden a aumentar las deformaciones totales para vigas.

Este trabajo presenta un modelo analítico de la curva elástica para vigas rectangulares con cartelas parabólicas, bajo carga concentrada y momentos en los extremos, considerando deformaciones por flexión y cortante (Teoría de Timoshenko) para obtener las rotaciones y desplazamientos, el enfoque principal de esta investigación. El modelo propuesto se divide en tres partes para la viga:

$$\begin{array}{rcl} 0 & \leq x \leq & a, \\ a & \leq x \leq & L-c, \\ L-c & \leq x \leq & L. \end{array}$$

El modelo actual de la curva elástica para vigas rectangulares bajo carga concentrada considera solo las deformaciones por flexión (Teoría de Euler-Bernoulli).

## 2. Modelo propuesto

La Figura 1 muestra la diferencia entre la teoría de Timoshenko y la teoría de Euler-Bernoulli. La primera teoría incluye el efecto de los esfuerzos de flexión y cortante sobre la deformación:

$$\frac{dy}{dx} = \frac{dy_c}{dx} + \frac{dy_f}{dx}. \tag{1}$$

Y la segunda teoría incluye el efecto de los esfuerzos de flexión en la deformación [22, 23]:

$$\frac{dy}{dx} = \frac{dy_f}{dx}. \tag{2}$$

La teoría de Timoshenko considera las deformaciones por flexión y cortante, esto es válido para los miembros cortos y largos. La ecuación de la curva elástica se presenta de la siguiente manera [22, 23]:

$$\frac{d^2y}{dx^2} = \frac{d^2y_c}{dx^2} + \frac{d^2y_f}{dx^2}, \tag{3}$$

$$\frac{d^2y}{dx^2} = -\left(\frac{1}{G\,A_{cx}}\right)\frac{dV_x}{dx} - \frac{M_z}{E\,I_z}, \tag{4}$$

donde:

$G =$ Módulo de cortante.
$y =$ Desplazamiento total.
$y_c =$ Desplazamiento por cortante debido a la fuerza cortante.

**Tabla 1.** Propiedades de la sección rectangular con cartelas parabólicas

| **Concepto** | **Ecuaciones** | | |
|---|---|---|---|
| **Intervalo** | $0 \leq x \leq a$ | $a \leq x \leq L-c$ | $L-c \leq x \leq L$ |
| $h_x(x)$ | $\dfrac{a^2h+u\left(x-a\right)^2}{a^2}$ | $h$ | $\dfrac{c^2h+s\left(x-L+c\right)^2}{c^2}$ |
| $A_{cx}(x)$ | $\dfrac{5b\left[a^2h+u\left(x-a\right)^2\right]}{6a^2}$ | $\dfrac{5bh}{6}$ | $\dfrac{5b\left[c^2h+s\left(x-L+c\right)^2\right]}{6c^2}$ |
| $I_z(x)$ | $\dfrac{b\left[a^2h+u\left(x-a\right)^2\right]^3}{12a^6}$ | $\dfrac{bh^3}{12}$ | $\dfrac{b\left[c^2h+s\left(x-L+c\right)^2\right]^3}{12c^6}$ |

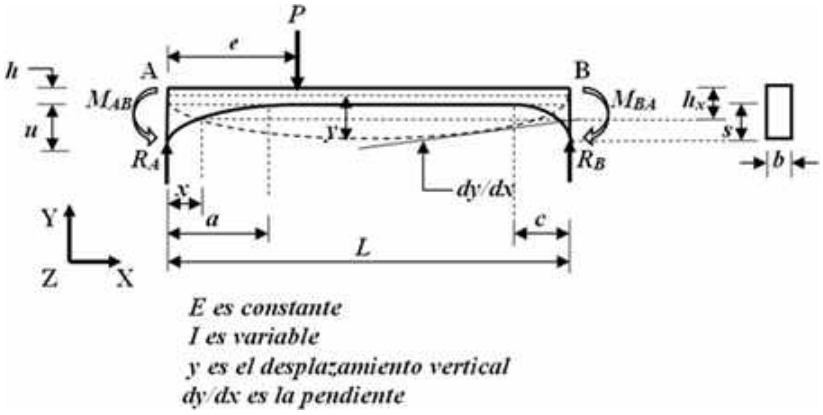



**Fig. 2.** Viga sometida a un momento en cada extremo y una carga concentrada

$y_f$ = Desplazamiento por flexión debido al momento.
$A_{cx}$ = Área de cortante.
$E$ = Módulo de elasticidad.
$M_z$ = Momento de flexión alrededor del eje $Z$.
$I_z$ = Momento de inercia alrededor del eje $Z$.

Las rotaciones en cualquier parte de la viga por integración de la ecuación (4) se obtienen:

$$\frac{dy}{dx} = -\frac{V_x}{G\,A_{cx}} - \int \frac{M_z}{E\,I_z}\,dx. \tag{5}$$

En la Figura 2 se muestra la viga AB sometida a un momento en cada extremo y una carga concentrada, así como su sección transversal rectangular considerando que el ancho "$b$" es constante y la altura "$h_x$" es variable de forma parabólica en sus extremos y constante en la parte central.

Los valores de las reacciones en los apoyos $R_A$ y $R_B$ se obtienen de las siguientes ecuaciones:

$$R_A = \frac{M_{AB}-M_{BA}}{L} + \frac{P(L-e)}{L}, \tag{6}$$

$$R_B = \frac{Pe}{L} - \frac{M_{AB}-M_{BA}}{L}. \tag{7}$$

Las fuerzas cortantes y momentos alrededor del eje $Z$ a una distancia x en cualquier parte de la trabe son:

$$V_x = R_A \rightarrow (0 \leq x \leq e), \tag{8}$$

$$V_x = R_A - P \rightarrow (e \leq x \leq L), \tag{9}$$

$$M_z = R_A\,x - M_{AB} \rightarrow (0 \leq x \leq e), \tag{10}$$

$$M_z = (R_A - P)x + Pe - M_{AB} \rightarrow (e \leq x \leq L). \tag{11}$$

Sustituyendo las ecuaciones (8) y (10) en la ecuación (5), para $0 \leq x \leq e$ se obtiene:

$$\frac{dy}{dx} = \frac{1}{E} \int \frac{R_A\, x - M_{AB}}{I_z}\, dx + \frac{R_A}{G\, A_{cx}}. \tag{12}$$

Sustituyendo las ecuaciones (9) y (11) en la ecuación (5), para $e \leq x \leq L$ se obtiene:

$$\frac{dy}{dx} = \frac{1}{E} \int \frac{(R_A - P)x + Pe - M_{AB}}{I_z}\, dx + \frac{R_A - P}{G\, A_{cx}}. \tag{13}$$

La Tabla 1 muestra las ecuaciones de las alturas "$h_x(x)$" a una distancia "$x$", áreas de cortante "$A_{cx}(x)$" a una distancia "$x$", y el momento de inercia "$I_z(x)$" alrededor del eje "$z$" a una distancia "$x$" para cada intervalo.

### 2.1. Cuando la carga se localiza de $0 \leq x \leq a$

Sustituyendo las propiedades de la Tabla 1 en las ecuaciones (12) y (13) para cada intervalo, y resolviendo las integrales se obtienen las rotaciones en cualquier lugar ($dy/dx$), y posteriormente se desarrolla la integral para obtener los desplazamientos en cualquier lugar de la viga ($y$). Las ecuaciones simplificadas para intervalo se muestran a continuación, para $0 \leq x \leq e$:

$$\begin{aligned}\frac{dy}{dx} \;=\; & -\frac{3a^2\left\{2R_A a^4 h^2 + (R_A a - M_{AB})\, u\,(a-x)\left[3u(a-x)^2 + 5a^2 h\right]\right\}}{2Ebh^2 u\left[a^2 h + u(a-x)^2\right]^2} - \\ & \frac{9a\,(R_A a - M_{AB})}{2Ebh^2\sqrt{hu}} \arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right] + \frac{6R_A a^2}{5Gb\left[a^2 h + u\,(a-x)^2\right]} + C_1,\end{aligned} \tag{14}$$

$$\begin{aligned}y \;=\; & \left[\frac{3a\left[3u\left((R_A a - M_{AB})(a-x) + R_A a^2 h\right)\right]}{2Ebh^2 u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right] \arctan\left(\frac{u(a-x)}{a\sqrt{hu}}\right) + \\ & \frac{9a^2\,(R_A a - M_{AB})}{4Ebh^2 u} \ln\left(a^2 h\right) - \frac{3a^4\,(R_A x - M_{AB})}{2Ebhu\,(a^2 h + u(a-x)^2)} + C_1\, x + C_2.\end{aligned} \tag{15}$$

Para $e \leq x \leq a$:

$$\begin{aligned}\frac{dy}{dx} \;=\; & -\frac{3a^2\,(a-x)\left[(R_A - P)\,a + Pe - M_{AB}\right]\left[5a^2 h + 3u\,(a-x)^2\right]}{2Ebh^2\left[a^2 h + u\,(a-x)^2\right]^2} - \frac{3a^6\,(R_A - P)}{Ebu\left[a^2 h + u\,(a-x)^2\right]^2} - \\ & \frac{9a\left[(R_A - P)\,a + Pe - M_{AB}\right]}{2Ebh^2\sqrt{hu}} \arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right] + \frac{6\,(R_A - P)\,a^2}{5Gb\left[a^2 h + u\,(a-x)^2\right]} + C_3,\end{aligned} \tag{16}$$

$$\begin{aligned}y \;=\; & \left[\frac{3a\left\{3u(a-x)\left[(R_A - P)a + Pe - M_{AB}\right] + (R_A - P)a^2 h\right\}}{2Ebh^2 u\sqrt{hu}} - \frac{6(R_A - P)a}{5Gb\sqrt{hu}}\right] \arctan\left(\frac{u(a-x)}{a\sqrt{hu}}\right) + \\ & \frac{9a^2\left[(R_A - P)\,a + Pe - M_{AB}\right]}{4Ebh^2 u} \ln\left(a^2 h\right) - \frac{3a^4\left[(R_A - P)\,x + Pe - M_{AB}\right]}{2Ebhu\left[a^2 h + u\,(a-x)^2\right]} + C_3\, x + C_4.\end{aligned} \tag{17}$$

Para $a \leq x \leq L - c$:

$$\frac{dy}{dx} = \frac{6x\left[(R_A - P)\,x + 2\,(Pe - M_{AB})\right]}{Ebh^3} + \frac{6\,(R_A - P)}{5Gbh} + C_5, \tag{18}$$

$$y = \frac{2x^2\left[(R_A - P)\,x + 3\,(Pe - M_{AB})\right]}{Ebh^3} + \frac{6\,(R_A - P)\,x}{5Gbh} + C_5\, x + C_6. \tag{19}$$

Para $L-c \leq x \leq L$:

$$\frac{dy}{dx} = -\frac{3c^2\left\{2(R_A-P)c^4h^2+\left[(R_A-P)(L-c)+Pe-M_{AB}\right]s(L-c-x)\left[3s(L-c-x)^2+5c^2h\right]\right\}}{2Ebh^2s\left(c^2h+s(L-c-x)^2\right)^2} - \frac{9c\left[(R_A-P)(L-c)+Pe-M_{AB}\right]}{2Ebh^2\sqrt{hs}}\arctan\left(\frac{s(L-c-x)}{c\sqrt{hs}}\right)+\frac{6(R_A-P)c^2}{5Gb\left[c^2h+s(L-c-x)^2\right]}+C_7, \tag{20}$$

$$y = \left[\frac{9c\left[(R_A-P)(L-c)+Pe-M_{AB}\right](L-c-x)}{2Ebh^2\sqrt{hs}}+\frac{3(R_A-P)c^3}{2Ebhs\sqrt{hs}}-\frac{6(R_A-P)c}{5Gb\sqrt{hs}}\right]\arctan\left(\frac{s(L-c-x)}{c\sqrt{hs}}\right)+ \frac{9c^2\left[(R_A-P)(L-c)+Pe-M_{AB}\right]}{4Ebh^2s}\ln\left(c^2h\right)-\frac{3c^4\left[(R_A-P)x+Pe-M_{AB}\right]}{2Ebhs\left[c^2h+s(L-c-x)^2\right]}+C_7\,x+C_8. \tag{21}$$

Las ocho condiciones conocidas que debe cumplir la viga para obtener las constantes de integración son:

1. Sustituyendo la condición $x=0$ e $y=0$ en la ecuación (15).
2. Sustituyendo la condición $x=e$ en las ecuaciones (14) y (16), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
3. Sustituyendo la condición $x=e$ en las ecuaciones (15) y (17), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
4. Sustituyendo la condición $x=a$ en las ecuaciones (16) y (18), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
5. Sustituyendo la condición $x=a$ en las ecuaciones (17) y (19), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
6. Sustituyendo la condición $x=L-c$ en las ecuaciones (18) y (20), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
7. Sustituyendo la condición $x=L-c$ en las ecuaciones (19) y (21), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
8. Sustituyendo la condición $x=L$ e $y=0$ en la ecuación (21).

Ahora, sustituyendo las ocho condiciones conocidas en las ecuaciones correspondientes, se generan ocho ecuaciones para obtener las constantes de integración. Las constantes de integración se muestran a continuación:

$$C_1 = -\frac{6Pa^2(L-e)}{5GbL\left[a^2h+u(a-e)^2\right]}-\frac{6(R_A-P)(L-a-c)}{5GbLh}+\left[\frac{3a^2\left[3u(R_Aa-M_{AB})+R_Aah\right]}{2EbLh^2u\sqrt{hu}}- \frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]+\left[\frac{3Pa\left[3u(a-e)(L-a)-a^2h\right]}{2EbLh^2u\sqrt{hu}}+\frac{6Pa}{5GbL\sqrt{hu}}\right]\arctan\left[\frac{u(a-e)}{a\sqrt{hu}}\right]+ \frac{3c^2\left[(R_A-P)L+Pe-M_{AB}\right]}{2EbLhs(h+s)}+\left[\frac{3c^2\left\{(R_A-P)\left[ch-3s(L-c)\right]-3s(Pe-M_{AB})\right\}}{2EbLh^2s\sqrt{hs}}- \frac{6(R_A-P)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right]+\frac{3M_{AB}a^2}{2EbLhu(h+u)}+\frac{3Pa^2(L-e)\left\{2a^4h^2+u(a-e)^2\left[3u(a-e)^2+5a^2h\right]\right\}}{2EbLh^2u\left[a^2h+u(a-e)^2\right]^2}+ \frac{a^2(R_A-P)\left[4u(3L-2a)+3h(2L-a)\right]+3a(Pe-M_{AB})\left[4u(2L-a)+ah\right]}{2EbLh^3u}- \frac{(R_A-P)\left[3c^2h(L+c)+4s(L-c)^2(L+2c)\right]+3(Pe-M_{AB})\left[c^2h+4s\left(L^2-c^2\right)\right]}{2EbLh^3s}, \tag{22}$$

$$C_2 = -\left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] - \frac{9a^2\left(R_A a - M_{AB}\right)}{4Ebh^2u}\ln\left(a^2h\right) - \frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}, \tag{23}$$

$$\begin{aligned} C_3 = {} & \frac{6Pa^2e}{5GbL\left[a^2h + u\left(a-e\right)^2\right]} - \frac{6\left(R_A - P\right)\left(L - a - c\right)}{5GbLh} + \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2EbLh^2u\sqrt{hu}} - \right. \\ & \left. \frac{6R_A a}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] - \left[\frac{3Pa^2\left[ah + 3u\left(a-e\right)\right]}{2EbLh^2u\sqrt{hu}} - \frac{6Pa}{5GbL\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right] + \\ & \frac{3c^2\left[\left(R_A - P\right)L + Pe - M_{AB}\right]}{2EbLhs\left(h+s\right)} + \left[\frac{3c^2\left\{\left(R_A - P\right)\left[ch - 3s\left(L-c\right)\right] - 3s\left(Pe - M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \right. \\ & \left. \frac{6\left(R_A - P\right)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right] + \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} - \frac{3Pa^2e\left\{2a^4h^2 + u\left(a-e\right)^2\left[3u\left(a-e\right)^2 + 5a^2h\right]\right\}}{2EbLh^2u\left[a^2h + u\left(a-e\right)^2\right]^2} + \\ & \frac{a^2\left(R_A - P\right)\left[4u\left(3L - 2a\right) + 3h\left(2L - a\right)\right] + 3a\left(Pe - M_{AB}\right)\left[4u\left(2L - a\right) + ah\right]}{2EbLh^3u} - \\ & \frac{\left(R_A - P\right)\left[3c^2h\left(L + c\right) + 4s\left(L - c\right)^2\left(L + 2c\right)\right] + 3\left(Pe - M_{AB}\right)\left[c^2h + 4s\left(L^2 - c^2\right)\right]}{2EbLh^3s}, \end{aligned} \tag{24}$$

$$\begin{aligned} C_4 = {} & -\frac{6Pa^2e}{5Gb\left[a^2h + u\left(a-e\right)^2\right]} - \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] - \\ & \frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)} + \left[\frac{3Pa^2\left[ah + 3u\left(a-e\right)\right]}{2Ebh^2u\sqrt{hu}} - \frac{6Pa}{5Gb\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right] - \\ & \frac{9a^2\left[\left(R_A - P\right)a + Pe - M_{AB}\right]}{4Ebh^2u}\ln\left(a^2h\right) + \frac{3Pa^2e\left\{2a^4h^2 + u\left(a-e\right)^2\left[3u\left(a-e\right)^2 + 5a^2h\right]\right\}}{2Ebh^2u\left[a^2h + u\left(a-e\right)^2\right]^2}, \end{aligned} \tag{25}$$

$$\begin{aligned} C_5 = {} & \frac{6Pa^2e}{5GbL\left[a^2h + u\left(a-e\right)^2\right]} - \frac{6\left(R_A - P\right)\left(L - a - c\right)}{5GbLh} + \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_A a}{5GbL\sqrt{hu}}\right] \\ & \arctan\left[\sqrt{\frac{u}{h}}\right] + \left[\frac{3c^2\left\{\left(R_A - P\right)\left[ch - 3s\left(L-c\right)\right] - 3s\left(Pe - M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\left(R_A - P\right)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right] - \\ & \left[\frac{3Pa^2\left[ah + 3u\left(a-e\right)\right]}{2EbLh^2u\sqrt{hu}} - \frac{6Pa}{5GbL\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right] + \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} - \\ & \frac{3Pa^2e\left\{2a^4h^2 + u\left(a-e\right)^2\left[3u\left(a-e\right)^2 + 5a^2h\right]\right\}}{2EbLh^2u\left[a^2h + u\left(a-e\right)^2\right]^2} + \\ & \frac{3c^2\left[\left(R_A - P\right)L + Pe - M_{AB}\right]}{2EbLhs\left(h+s\right)} - \frac{a^3\left(R_A - P\right)\left(3h + 8u\right) + 3a^2\left(Pe - M_{AB}\right)\left(4u - h\right)}{2EbLh^3u} - \\ & \frac{\left(R_A - P\right)\left[3c^2h\left(L + c\right) + 4s\left(L - c\right)^2\left(L + 2c\right)\right] + 3\left(Pe - M_{AB}\right)\left[c^2h + 4s\left(L^2 - c^2\right)\right]}{2EbLh^3s}, \end{aligned} \tag{26}$$

$$C_6 = -\frac{6Pa^2e}{5Gb\left[a^2h+u\left(a-e\right)^2\right]}-\frac{6\left(R_A-P\right)a}{5Gbh}-\left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]$$
$$\arctan\left[\sqrt{\frac{u}{h}}\right]+\left[\frac{3Pa^2\left[ah+3u\left(a-e\right)\right]}{2Ebh^2u\sqrt{hu}}-\frac{6Pa}{5Gb\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right]-\frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}+$$
$$\frac{3Pa^2e\left\{2a^4h^2+u\left(a-e\right)^2\left[3u\left(a-e\right)^2+5a^2h\right]\right\}}{2Ebh^2u\left[a^2h+u\left(a-e\right)^2\right]^2}+\frac{a^3\left(R_A-P\right)\left(3h+8u\right)+3a^2\left(Pe-M_{AB}\right)\left(4u-h\right)}{2Ebh^3u}, \tag{27}$$

$$C_7 = \frac{6Pa^2e}{5GbL\left[a^2h+u\left(a-e\right)^2\right]}-\frac{6\left(R_A-P\right)\left(L-a-c\right)}{5GbLh}+\left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2EbLh^2u\sqrt{hu}}-\frac{6R_Aa}{5GbL\sqrt{hu}}\right]$$
$$\arctan\left[\sqrt{\frac{u}{h}}\right]-\left[\frac{3Pa^2\left[ah+3u\left(a-e\right)\right]}{2EbLh^2u\sqrt{hu}}-\frac{6Pa}{5GbL\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right]+\frac{3c^2\left[\left(R_A-P\right)L+Pe-M_{AB}\right]}{2EbLhs\left(h+s\right)}+$$
$$\left[\frac{3c^2\left\{\left(R_A-P\right)\left[ch-3s\left(L-c\right)\right]-3s\left(Pe-M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}}-\frac{6\left(R_A-P\right)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right]+\frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)}-$$
$$\frac{3Pa^2e\left\{2a^4h^2+u\left(a-e\right)^2\left[3u\left(a-e\right)^2+5a^2h\right]\right\}}{2EbLh^2u\left[a^2h+u\left(a-e\right)^2\right]^2}-\frac{a^3\left(R_A-P\right)\left(3h+8u\right)+3a^2\left(Pe-M_{AB}\right)\left(4u-h\right)}{2EbLh^3u}+$$
$$\frac{\left(R_A-P\right)\left(L-c\right)\left[3c^2h+8s\left(L-c\right)^2\right]-3\left(Pe-M_{AB}\right)\left[c^2h-4s\left(L-c\right)^2\right]}{2EbLh^3s}, \tag{28}$$

$$C_8 = \frac{6\left(R_A-P\right)\left(L-a-c\right)}{5Gbh}-\frac{6Pa^2e}{5Gb\left[a^2h+u\left(a-e\right)^2\right]}-\left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]$$
$$\arctan\left[\sqrt{\frac{u}{h}}\right]+\left[\frac{3Pa^2\left[ah+3u\left(a-e\right)\right]}{2Ebh^2u\sqrt{hu}}-\frac{6Pa}{5Gb\sqrt{hu}}\right]\arctan\left[\frac{u\left(a-e\right)}{a\sqrt{hu}}\right]-\frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}-$$
$$\frac{9c^2\left[\left(R_A-P\right)\left(L-c\right)+Pe-M_{AB}\right]}{4Ebh^2s}\ln\left(c^2h\right)+\frac{3Pa^2e\left\{2a^4h^2+u\left(a-e\right)^2\left[3u\left(a-e\right)^2+5a^2h\right]\right\}}{2Ebh^2u\left[a^2h+u\left(a-e\right)^2\right]^2}+$$
$$\frac{a^3\left(R_A-P\right)\left(3h+8u\right)+3a^2\left(Pe-M_{AB}\right)\left(4u-h\right)}{2Ebh^3u}-$$
$$\frac{\left(R_A-P\right)\left(L-c\right)\left[3c^2h+8s\left(L-c\right)^2\right]-3\left(Pe-M_{AB}\right)\left[c^2h-4s\left(L-c\right)^2\right]}{2Ebh^3s}. \tag{29}$$

### 2.2. Carga localizada de $a \leq x \leq L-c$

Sustituyendo las propiedades de la Tabla 1 en las ecuaciones (12) y (13) para cada intervalo, y resolviendo las integrales se obtienen las rotaciones en cualquier lugar $(dy/dx)$, y posteriormente se desarrolla la integral para obtener los desplazamientos en cualquier lugar de la viga $(y)$.

Las ecuaciones simplificadas para intervalo se muestran a continuación. Para $0 \leq x \leq a$:

$$\frac{dy}{dx} = -\frac{3a^2\left\{2R_A a^4h^2 + (R_A a - M_{AB})\,u\,(a-x)\left[3u\,(a-x)^2 + 5a^2h\right]\right\}}{2Ebh^2u\left[a^2h + u\,(a-x)^2\right]^2} - \frac{9a\,(R_A a - M_{AB})}{2Ebh^2\sqrt{hu}}\arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right] + \frac{6R_A a^2}{5Gb\left[a^2h + u\,(a-x)^2\right]} + C_1, \tag{30}$$

$$y = \left[\frac{3a\left[3u\,(R_A a - M_{AB})\,(a-x) + R_A a^2h\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right] + \frac{9a^2\,(R_A a - M_{AB})}{4Ebh^2u}\ln\left(a^2h\right) - \frac{3a^4\,(R_A x - M_{AB})}{2Ebhu\left[a^2h + u\,(a-x)^2\right]} + C_1\,x + C_2. \tag{31}$$

Para $a \le x \le e$:

$$\frac{dy}{dx} = \frac{6x\,(R_A x - 2M_{AB})}{Ebh^3} + \frac{6R_A}{5Gbh} + C_3, \tag{32}$$

$$y = \frac{2x^2\,(R_A x - 3M_{AB})}{Ebh^3} + \frac{6R_A x}{5Gbh} + C_3\,x + C_4. \tag{33}$$

Para $e \le x \le L - c$:

$$\frac{dy}{dx} = \frac{6x\left[(R_A - P)\,x + 2\,(Pe - M_{AB})\right]}{Ebh^3} + \frac{6\,(R_A - P)}{5Gbh} + C_5, \tag{34}$$

$$y = \frac{2x^2\left[(R_A - P)\,x + 3\,(Pe - M_{AB})\right]}{Ebh^3} + \frac{6\,(R_A - P)\,x}{5Gbh} + C_5\,x + C_6. \tag{35}$$

Para $L - c \le x \le L$:

$$\frac{dy}{dx} = -\frac{3c^2\left\{2\,(R_A - P)\,c^4h^2 + \left[(R_A - P)\,(L-c) + Pe - M_{AB}\right]s\,(L-c-x)\left[3s\,(L-c-x)^2 + 5c^2h\right]\right\}}{2Ebh^2s\left[c^2h + s\,(L-c-x)^2\right]^2} - \frac{9c\left((R_A - P)\,(L-c) + Pe - M_{AB}\right)}{2Ebh^2\sqrt{hs}}\arctan\left[\frac{s\,(L-c-x)}{c\sqrt{hs}}\right] + \frac{6\,(R_A - P)\,c^2}{5Gb\left[c^2h + s\,(L-c-x)^2\right]} + C_7, \tag{36}$$

$$y = \left[\frac{9c\left[(R_A - P)\,(L-c) + Pe - M_{AB}\right](L-c-x)}{2Ebh^2\sqrt{hs}} + \frac{3c^3\,(R_A - P)}{2Ebhs\sqrt{hs}} - \frac{6\,(R_A - P)\,c}{5Gb\sqrt{hs}}\right]\arctan\left[\frac{s\,(L-c-x)}{c\sqrt{hs}}\right] + \frac{9c^2\left[(R_A - P)\,(L-c) + Pe - M_{AB}\right]}{4Ebh^2s}\ln\left(c^2h\right) - \frac{3c^4\left((R_A - P)\,x + Pe - M_{AB}\right)}{2Ebhs\left[c^2h + s\,(L-c-x)^2\right]} + C_7\,x + C_8. \tag{37}$$

Las ocho condiciones conocidas que debe cumplir la viga para obtener las constantes de integración son:

1. Sustituyendo la condición $x = 0$ e $y = 0$ en la ecuación (31).
2. Sustituyendo la condición $x = a$ en las ecuaciones (30) y (32), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
3. Sustituyendo la condición $x = a$ en las ecuaciones (31) y (33), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
4. Sustituyendo la condición $x = e$ en las ecuaciones (32) y (34), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
5. Sustituyendo la condición $x = e$ en las ecuaciones (33) y (35), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
6. Sustituyendo la condición $x = L - c$ en las ecuaciones (34) y (36), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
7. Sustituyendo la condición $x = L - c$ en las ecuaciones (35) y (37), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
8. Sustituyendo la condición $x = L$ e $y = 0$ en la ecuación (37).

Ahora, sustituyendo las ocho condiciones conocidas en las ecuaciones correspondientes, se generan ocho ecuaciones para obtener las constantes de integración. Las constantes de integración se muestran a continuación:

$$
\begin{aligned}
C_1 = {} & -\frac{6Pc}{5GbLh} - \frac{6R_A\left(L-a-c\right)}{5GbLh}\left[\frac{3c^2\left\{\left(R_A-P\right)\left[ch-3s\left(L-c\right)\right]-3s\left(Pe-M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\left(R_A-P\right)c}{5GbL\sqrt{hs}}\right] \\
& \arctan\left[\sqrt{\frac{s}{h}}\right] + \frac{3c^2\left[\left(R_A-P\right)L+Pe-M_{AB}\right]}{2EbLh^2s\left(h+s\right)} + \left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_Aa}{5GbL\sqrt{hu}}\right] \\
& \arctan\left[\sqrt{\frac{u}{h}}\right] + \frac{3Pa^2\left(L-e\right)\left\{2a^4h^2+u\left(a-e\right)^2\left[3u\left(a-e\right)^2+5a^2h\right]\right\}}{2EbLh^2u\left[a^2h+u\left(a-e\right)^2\right]^2} + \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} - \\
& \frac{2Pe^2\left(3L-e\right)}{EbLh^3} - \frac{a\left\{R_Aa\left[a\left(3h+8u\right)-6L\left(h+2u\right)\right]+3M_{AB}\left[a\left(h-4u\right)+8Lu\right]\right\}}{2EbLh^3u} - \\
& \frac{\left(R_A-P\right)\left[3c^2h\left(L+c\right)+4s\left(L-c\right)^2\left(L+2c\right)\right]+3\left(Pe-M_{AB}\right)\left[c^2h+4s\left(L^2-c^2\right)\right]}{2EbLh^3s},
\end{aligned}
\tag{38}
$$

$$
C_2 = -\left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_Aa}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] - \frac{9a^2\left(R_Aa-M_{AB}\right)}{4Ebh^2u}\ln\left(a^2h\right) - \frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}, \tag{39}
$$

$$
\begin{aligned}
C_3 = {} & -\frac{6Pc}{5GbLh} - \frac{6R_A\left(L-a-c\right)}{5GbLh} + \left[\frac{3c^2\left\{\left(R_A-P\right)\left[ch-3s\left(L-c\right)\right]-3s\left(Pe-M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\left(R_A-P\right)c}{5GbL\sqrt{hs}}\right] \\
& \arctan\left[\sqrt{\frac{s}{h}}\right] + \left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] + \\
& \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} + \frac{2Pe^2\left(3L-e\right)}{EbLh^3} - \frac{a^2\left[R_Aa\left(3h+8u\right)+3M_{AB}\left(h-4u\right)\right]}{2EbLh^3u} + \frac{3c^2\left[\left(R_A-P\right)L+Pe-M_{AB}\right]}{2EbLh^2s\left(h+s\right)} - \\
& \frac{\left(R_A-P\right)\left[3c^2h\left(L+c\right)+4s\left(L-c\right)^2\left(L+2c\right)\right]+3\left(Pe-M_{AB}\right)\left[c^2h+4s\left(L^2-c^2\right)\right]}{2EbLh^3s},
\end{aligned}
\tag{40}
$$

$$C_4 = -\frac{6R_A a}{5Gbh} - \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] + \frac{a^2\left[R_A a\left(3h+8u\right) + 3M_{AB}\left(h-4u\right)\right]}{2EbLh^3u} - \frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}, \tag{41}$$

$$C_5 = \frac{6R_A a}{5GbLh} - \frac{6\left(R_A - P\right)\left(L-c\right)}{5GbLh} + \left[\frac{3c^2\left\{\left(R_A - P\right)\left[ch - 3s\left(L-c\right)\right] - 3s\left(Pe - M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\left(R_A - P\right)c}{5GbL\sqrt{hs}}\right]$$
$$\arctan\left[\sqrt{\frac{s}{h}}\right] + \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_A a}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] +$$
$$\frac{3c^2\left[\left(R_A - P\right)L + Pe - M_{AB}\right]}{2EbLh^2s\left(h+s\right)} + \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} + \frac{2Pe^2\left(3L-e\right)}{EbLh^3} - \frac{a^2\left[R_A a\left(3h+8u\right) + 3M_{AB}\left(h-4u\right)\right]}{2EbLh^3u} -$$
$$\frac{\left(R_A - P\right)\left[3c^2h\left(L+c\right) + 4s\left(L-c\right)^2\left(L+2c\right)\right] + 3\left(Pe - M_{AB}\right)\left[c^2h + 4s\left(L^2 - c^2\right)\right]}{2EbLh^3s}, \tag{42}$$

$$C_6 = -\frac{6R_A a}{5Gbh} - \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] - \frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)} +$$
$$\frac{2Pe^3}{Ebh^3} + \frac{a^3\left(R_A - P\right)\left(3h+8u\right) + 3a^2\left(Pe - M_{AB}\right)\left(4u-h\right)}{2Ebh^3u}, \tag{43}$$

$$C_7 = \frac{6R_A a}{5GbLh} - \frac{6\left(R_A - P\right)\left(L-c\right)}{5GbLh} + \left[\frac{3c^2\left\{\left(R_A - P\right)\left[ch - 3s\left(L-c\right)\right] - 3s\left(Pe - M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\left(R_A - P\right)c}{5GbL\sqrt{hs}}\right]$$
$$\arctan\left[\sqrt{\frac{s}{h}}\right] + \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_A a}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] +$$
$$\frac{3c^2\left[\left(R_A - P\right)L + Pe - M_{AB}\right]}{2EbLh^2s\left(h+s\right)} + \frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)} - \frac{2Pe^3}{EbLh^3} - \frac{a^2\left[R_A a\left(3h+8u\right) + 3M_{AB}\left(h-4u\right)\right]}{2EbLh^3u} +$$
$$\frac{\left(R_A - P\right)\left(L-c\right)\left[3c^2h + 8s\left(L-c\right)^2\right] - 3\left(Pe - M_{AB}\right)\left[c^2h - 4s\left(L-c\right)^2\right]}{2EbLh^3s}, \tag{44}$$

$$C_8 = -\frac{6R_A a}{5Gbh} + \frac{6\left(R_A - P\right)\left(L-c\right)}{5Gbh} - \left[\frac{3a^2\left[3u\left(R_A a - M_{AB}\right) + R_A ah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_A a}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] -$$
$$\frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)} + \frac{2Pe^3}{Ebh^3} + \frac{a^2\left[R_A a\left(3h+8u\right) + 3M_{AB}\left(h-4u\right)\right]}{2Ebh^3u} - \frac{9c^2\left[\left(R_A - P\right)\left(L-c\right) + Pe - M_{AB}\right]}{4Ebh^2s} \tag{45}$$
$$\ln\left(c^2h\right) - \frac{\left(R_A - P\right)\left(L-c\right)\left[3c^2h + 8s\left(L-c\right)^2\right] - 3\left(Pe - M_{AB}\right)\left[c^2h - 4s\left(L-c\right)^2\right]}{2EbLh^3s}.$$

### 2.3. Carga localizada de $L-c \leq x \leq L$

Sustituyendo las propiedades de la Tabla 1 en las ecuaciones (12) y (13) para cada intervalo, y resolviendo las integrales se obtienen las rotaciones en cualquier lugar $(dy/dx)$. Posteriormente se desarrolla la integral para obtener los desplazamientos en cualquier lugar de la viga $(y)$. Las ecuaciones simplificadas para intervalo se muestran a continuación. Para $0 \leq x \leq a$:

$$\frac{dy}{dx} = -\frac{3a^2\left\{2R_Aa^4h^2+(R_Aa-M_{AB})\,u\,(a-x)\left[3u\,(a-x)^2+5a^2h\right]\right\}}{2Ebh^2u\left[a^2h+u\,(a-x)^2\right]^2}- \frac{9a\,(R_Aa-M_{AB})}{2Ebh^2\sqrt{hu}}\arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right]+\frac{6R_Aa^2}{5Gb\left[a^2h+u\,(a-x)^2\right]}+C_1, \tag{46}$$

$$y = \left[\frac{3a\left[3u\,(R_Aa-M_{AB})\,(a-x)+R_Aa^2h\right]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]\arctan\left[\frac{u\,(a-x)}{a\sqrt{hu}}\right]+ \frac{9a^2\,(R_Aa-M_{AB})}{4Ebh^2u}\ln\left(a^2h\right)-\frac{3a^4\,(R_Ax-M_{AB})}{2Ebhu\left[a^2h+u\,(a-x)^2\right]}+C_1x+C_2. \tag{47}$$

Para $a \leq x \leq L-c$:

$$\frac{dy}{dx} = \frac{6x\,(R_Ax-2M_{AB})}{Ebh^3}+\frac{6R_A}{5Gbh}+C_3, \tag{48}$$

$$y = \frac{2x^2\,(R_Ax-3M_{AB})}{Ebh^3}+\frac{6R_Ax}{5Gbh}+C_3\,x+C_4. \tag{49}$$

Para $L-c \leq x \leq e$:

$$\frac{dy}{dx} = -\frac{3c^2\left\{2R_Ac^4h^2+[R_A\,(L-c)-M_{AB}]\,s\,(L-c-x)\left[3s\,(L-c-x)^2+5c^2h\right]\right\}}{2Ebh^2s\left[c^2h+s\,(L-c-x)^2\right]^2}- \frac{9c\,(R_A\,(L-c)-M_{AB})}{2Ebh^2\sqrt{hs}}\arctan\left[\frac{s\,(L-c-x)}{c\sqrt{hs}}\right]+\frac{6R_Ac^2}{5Gb\left[c^2h+s\,(L-c-x)^2\right]}+C_5, \tag{50}$$

$$y = \left[\frac{3c\left\{3s\,[R_A\,(L-c)-M_{AB}]\,(L-c-x)+R_Ac^2h\right\}}{2Ebh^2s\sqrt{hs}}-\frac{6R_Ac}{5Gb\sqrt{hs}}\right]\arctan\left[\frac{s\,(L-c-x)}{c\sqrt{hs}}\right]+ \frac{9c^2\,[R_A\,(L-c)-M_{AB}]}{4Ebh^2s}\ln\left(c^2h\right)-\frac{3c^4\,(R_Ax-M_{AB})}{2Ebhs\left[c^2h+s\,(L-c-x)^2\right]}+C_5x+C_6. \tag{51}$$

Para $e \leq x \leq L$:

$$\frac{dy}{dx} = -\frac{3c^2\left\{2\,(R_A-P)\,c^4h^2+[(R_A-P)\,(L-c)+Pe-M_{AB}]\,s\,(L-c-x)\left[3s\,(L-c-x)^2+5c^2h\right]\right\}}{2Ebh^2s\left[c^2h+s\,(L-c-x)^2\right]^2}- \frac{9c\,((R_A-P)\,(L-c)+Pe-M_{AB})}{2Ebh^2\sqrt{hs}}\arctan\left[\frac{s\,(L-c-x)}{c\sqrt{hs}}\right]+\frac{6\,(R_A-P)\,c^2}{5Gb\left[c^2h+s\,(L-c-x)^2\right]}+C_7, \tag{52}$$

$$y = \left[\frac{9c\left[\left(R_A-P\right)\left(L-c\right)+Pe-M_{AB}\right]\left(L-c-x\right)}{2Ebh^2\sqrt{hs}}+\frac{9c^3\left(R_A-P\right)}{2Ebh\sqrt{hs}}-\frac{6\left(R_A-P\right)c}{5Gb\sqrt{hs}}\right]\arctan\left[\frac{s\left(L-c-x\right)}{c\sqrt{hs}}\right]+$$

$$\frac{9c^2\left[\left(R_A-P\right)\left(L-c\right)+Pe-M_{AB}\right]}{4Ebh^2s}\ln\left(c^2h\right)-\frac{3c^4\left(\left(R_A-P\right)x+Pe-M_{AB}\right)}{2Ebhs\left[c^2h+s\left(L-c-x\right)^2\right]}+C_7\,x+C_8. \tag{53}$$

Las ocho condiciones conocidas que debe cumplir la viga para obtener las constantes de integración son:

1. Sustituyendo la condición $x = 0$ e $y = 0$ en la ecuación (47).
2. Sustituyendo la condición $x = a$ en las ecuaciones (46) y (48), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
3. Sustituyendo la condición $x = a$ en las ecuaciones (47) y (49), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
4. Sustituyendo la condición $x = L - c$ en las ecuaciones (48) y (50), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
5. Sustituyendo la condición $x = L - c$ en las ecuaciones (49) y (51), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
6. Sustituyendo la condición $x = e$ en las ecuaciones (50) y (52), estas dos ecuaciones se igualan, porque las rotaciones "$dy/dx$" en este punto deben ser las mismas.
7. Sustituyendo la condición $x = e$ en las ecuaciones (51) y (53), estas dos ecuaciones se igualan, porque los desplazamientos "$y$" en este punto deben ser los mismos.
8. Sustituyendo la condición $x = L$ e $y = 0$ en la ecuación (53).

Sustituyendo las condiciones en las ecuaciones correspondientes, se obtienen las siguientes: constantes de integración:

$$\begin{aligned}C_1 = {} & -\frac{6R_A\left(L-a-c\right)}{5GbLh}+\left[\frac{3Pc^2\left[3s\left(L-c-e\right)-ch\right]}{2EbLh^2s\sqrt{hs}}+\frac{6Pc}{5GbL\sqrt{hs}}\right]\arctan\left[\frac{s\left(L-c-e\right)}{c\sqrt{hs}}\right]+\frac{3M_{AB}a^2}{2EbLhu\left(h+u\right)}+\\ & \left[\frac{3c^2\left\{\left(R_A-P\right)\left[ch-3s\left(L-c\right)\right]-3s\left(Pe-M_{AB}\right)\right\}}{2EbLh^2s\sqrt{hs}}-\frac{6\left(R_A-P\right)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right]+\\ & \left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2EbLh^2u\sqrt{hu}}-\frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-\frac{6Pc^2\left(L-e\right)}{5GbL\left[c^2h+s\left(L-c-e\right)^2\right]}-\\ & \frac{a^2\left[R_Aa\left(3h+8u\right)+3M_{AB}\left(h-4u\right)\right]}{2EbLh^3u}+\frac{3a\left[R_AaL\left(h+2u\right)-4M_{AB}Lu\right]}{EbLh^3u}+\\ & \frac{3Pc^2\left(L-e\right)\left\{2c^4h^2+s\left(L-c-e\right)^2\left[3s\left(L-c-e\right)^2+5c^2h\right]\right\}}{2EbLh^2s\left[c^2h+s\left(L-c-e\right)^2\right]^2}+\frac{3c^2\left[\left(R_A-P\right)L+Pe-M_{AB}\right]}{2EbLh^2s\left(h+s\right)}-\\ & \frac{R_A\left[3c^2h\left(L+c\right)+4s\left(L-c\right)^2\left(L+2c\right)\right]-3M_{AB}\left[c^2h+4s\left(L^2-c^2\right)\right]}{2EbLh^3s},\end{aligned} \tag{54}$$

$$C_2 = -\left[\frac{3a^2\left[3u\left(R_Aa-M_{AB}\right)+R_Aah\right]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-\frac{9a^2\left(R_Aa-M_{AB}\right)}{4Ebh^2u}\ln\left(a^2h\right)-\frac{3M_{AB}a^2}{2Ebhu\left(h+u\right)}, \tag{55}$$

$$C_3 = -\frac{6R_A(L-a-c)}{5GbLh}+\left[\frac{3Pc^2[3s(L-c-e)-ch]}{2EbLh^2s\sqrt{hs}}+\frac{6Pc}{5GbL\sqrt{hs}}\right]\arctan\left[\frac{s(L-c-e)}{c\sqrt{hs}}\right]+\frac{3M_{AB}a^2}{2EbLhu(h+u)}+$$
$$\left[\frac{3c^2\{(R_A-P)[ch-3s(L-c)]-3s(Pe-M_{AB})\}}{2EbLh^2s\sqrt{hs}}-\frac{6(R_A-P)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right]+$$
$$\left[\frac{3a^2[3u(R_Aa-M_{AB})+R_Aah]}{2EbLh^2u\sqrt{hu}}-\frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-\frac{6Pc^2(L-e)}{5GbL\left[c^2h+s(L-c-e)^2\right]}+$$
$$\frac{3Pc^2(L-e)\left\{2c^4h^2+s(L-c-e)^2\left[3s(L-c-e)^2+5c^2h\right]\right\}}{2EbLh^2s\left[c^2h+s(L-c-e)^2\right]^2}+\frac{3c^2[(R_A-P)L+Pe-M_{AB}]}{2EbLh^2s(h+s)}-$$
$$\frac{R_A\left[3c^2h(L+c)+4s(L-c)^2(L+2c)\right]-3M_{AB}\left[c^2h+4s\left(L^2-c^2\right)\right]}{2EbLh^3s}-\frac{a^2[R_Aa(3h+8u)+3M_{AB}(h-4u)]}{2EbLh^3u}, \tag{56}$$

$$C_4 = -\frac{6R_Aa}{5Gbh}-\left[\frac{3a^2[3u(R_Aa-M_{AB})+R_Aah]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-$$
$$\frac{3M_{AB}a^2}{2Ebhu(h+u)}+\frac{a^2[R_Aa(3h+8u)+3M_{AB}(h-4u)]}{2EbLh^3u}, \tag{57}$$

$$C_5 = -\frac{6R_A(L-a-c)}{5GbLh}+\left[\frac{3Pc^2[3s(L-c-e)-ch]}{2EbLh^2s\sqrt{hs}}+\frac{6Pc}{5GbL\sqrt{hs}}\right]\arctan\left[\frac{s(L-c-e)}{c\sqrt{hs}}\right]+\frac{3M_{AB}a^2}{2EbLhu(h+u)}+$$
$$\left[\frac{3c^2\{(R_A-P)[ch-3s(L-c)]-3s(Pe-M_{AB})\}}{2EbLh^2s\sqrt{hs}}-\frac{6(R_A-P)c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right]+$$
$$\left[\frac{3a^2[3u(R_Aa-M_{AB})+R_Aah]}{2EbLh^2u\sqrt{hu}}-\frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-\frac{6Pc^2(L-e)}{5GbL\left[c^2h+s(L-c-e)^2\right]}-$$
$$\frac{a^2[R_Aa(3h+8u)+3M_{AB}(h-4u)]}{2EbLh^3u}+\frac{2(L-c)^2[2R_A(L-c)-3M_{AB}]}{EbLh^3}+\frac{3c^2[R_A(L-c)+M_{AB}]}{2EbLh^2s}+$$
$$\frac{3c^2[(R_A-P)L+Pe-M_{AB}]}{2EbLh^2s(h+s)}+\frac{3Pc^2(L-e)\left\{2c^4h^2+s(L-c-e)^2\left[3s(L-c-e)^2+5c^2h\right]\right\}}{2EbLh^2s\left[c^2h+s(L-c-e)^2\right]^2}, \tag{58}$$

$$C_6 = \frac{6R_A(L-a-c)}{5Gbh}-\left[\frac{3a^2[3u(R_Aa-M_{AB})+R_Aah]}{2Ebh^2u\sqrt{hu}}-\frac{6R_Aa}{5Gb\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right]-$$
$$\frac{2(L-c)^2[2R_A(L-c)-3M_{AB}]}{Ebh^3}-\frac{3c^2[R_A(L-c)+M_{AB}]}{2EbLh^2s}+\frac{a^2[R_Aa(3h+8u)+3M_{AB}(4u-h)]}{2Ebh^3u}-$$
$$\frac{9c^2[R_A(L-c)-M_{AB}]}{4Ebh^2s}\ln\left(c^2h\right)-\frac{3M_{AB}a^2}{2Ebhu(h+u)}, \tag{59}$$

$$
\begin{aligned}
C_7 \;=\; & -\frac{6R_A\,(L-a-c)}{5GbLh} - \left[\frac{3Pc\left[3s\,(L-c-e)\,(L-c)+c^2h\right]}{2EbLh^2s\sqrt{hs}} - \frac{6Pc}{5GbL\sqrt{hs}}\right]\arctan\left[\frac{s\,(L-c-e)}{c\sqrt{hs}}\right] + \\
& \frac{3M_{AB}a^2}{2EbLhu\,(h+u)} + \left[\frac{3c^2\left\{(R_A-P)\,[ch-3s\,(L-c)]-3s\,(Pe-M_{AB})\right\}}{2EbLh^2s\sqrt{hs}} - \frac{6\,(R_A-P)\,c}{5GbL\sqrt{hs}}\right]\arctan\left[\sqrt{\frac{s}{h}}\right] + \\
& \frac{3c^2\left[(R_A-P)\,L+Pe-M_{AB}\right]}{2EbLh^2s\,(h+s)} + \left[\frac{3a^2\left[3u\,(R_Aa-M_{AB})+R_Aah\right]}{2EbLh^2u\sqrt{hu}} - \frac{6R_Aa}{5GbL\sqrt{hu}}\right]\arctan\left[\sqrt{\frac{u}{h}}\right] + \\
& \frac{6Pc^2e}{5GbL\left[c^2h+s\,(L-c-e)^2\right]} + \frac{2\,(L-c)^2\left[2R_A\,(L-c)-3M_{AB}\right]}{EbLh^3} + \frac{3c^2\left[R_A\,(L-c)+M_{AB}\right]}{2EbLh^2s} - \\
& \frac{a^2\left[R_Aa\,(3h+8u)+3M_{AB}\,(h-4u)\right]}{2EbLh^3u} - \frac{3Pc^2e\left\{2c^4h^2+s\,(L-c-e)^2\left[3s\,(L-c-e)^2+5c^2h\right]\right\}}{2EbLh^2s\left[c^2h+s\,(L-c-e)^2\right]^2},
\end{aligned}
\tag{60}
$$

$$
\begin{aligned}
C_8 \;=\; & \frac{6\,(R_A-P)\,(L-c)}{5Gbh} - \frac{6Pc^2e}{5Gb\left[c^2h+s\,(L-c-e)^2\right]} - \left[\frac{3a^2\left[3u\,(R_Aa-M_{AB})+R_Aah\right]}{2Ebh^2u\sqrt{hu}} - \frac{6R_Aa}{5Gb\sqrt{hu}}\right] \\
& \arctan\left[\frac{s\,(L-c-e)}{c\sqrt{hs}}\right] - \frac{3c^2\left[R_A\,(L-c)+M_{AB}\right]}{2Ebh^2s} + \frac{a^2\left[R_Aa\,(3h+8u)+3M_{AB}\,(h-4u)\right]}{2Ebh^3u} + \\
& \frac{3Pc^2e\left\{2c^4h^2+s\,(L-c-e)^2\left[3s\,(L-c-e)^2+5c^2h\right]\right\}}{2Ebh^2s\left[c^2h+s\,(L-c-e)^2\right]^2} - \frac{2\,(L-c)^2\left[2R_A\,(L-c)-3M_{AB}\right]}{Ebh^3} - \\
& \frac{9c^2\left[(R_A-P)\,(L-c)+Pe-M_{AB}\right]}{4Ebh^2s}\ln\left(c^2h\right) - \frac{3M_{AB}a^2}{2Ebhu\,(h+u)}.
\end{aligned}
\tag{61}
$$

## 3. Aplicación del modelo propuesto

Una manera de verificar la continuidad del modelo propuesto es como siguiente:

1. Sustituyendo el valor de "$x = a$" en las ecuaciones (16) y (18) (Cuando la carga se localiza de $e \leq x \leq a$), en la ecuación (30) y (32) (Cuando la carga se localiza de $a \leq x \leq L-c$), en la ecuación (46) y (48) (Cuando la carga se localiza de $L-c \leq x \leq L$), las rotaciones obtenidas de las dos ecuaciones son iguales.

2. Sustituyendo el valor de "$x = L-c$" en las ecuaciones (18) y (20) (Cuando la carga se localiza de $0 \leq x \leq a$), en la ecuación (34) y (36) (Cuando la carga se localiza de $a \leq x \leq L-c$), en la ecuación (48) y (50) (Cuando la carga se localiza de $L-c \leq x \leq e$), se obtienen las mismas rotaciones en ambos casos.

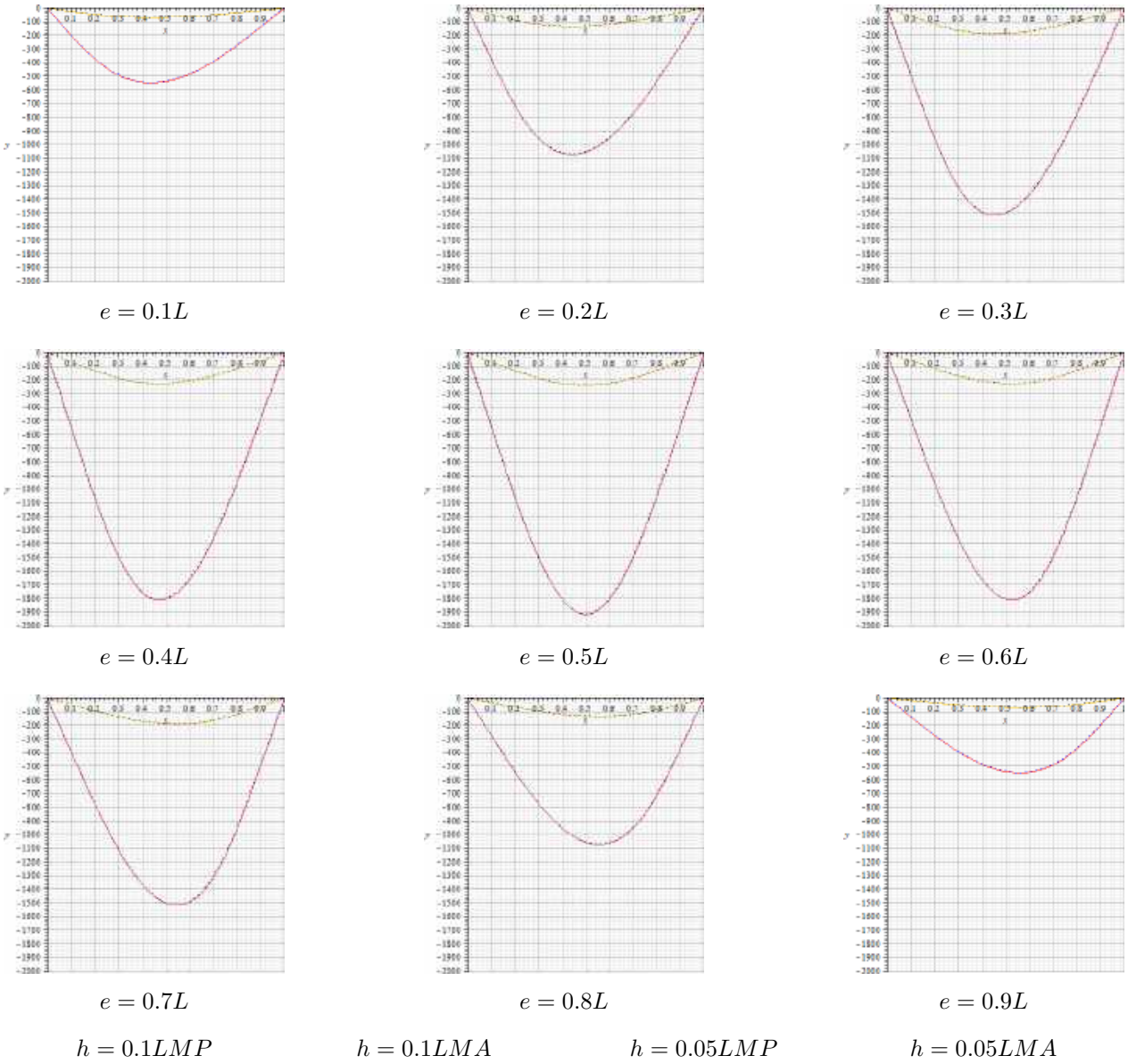
3. Sustituyendo el valor de "$x = e$" en las ecuaciones (14) y (16) (Cuando la carga se localiza de $0 \leq x \leq a$), en la ecuación (32) y (34) (Cuando la carga se localiza de $a \leq x \leq L-c$), en la ecuación (50) y (52) (Cuando la carga se localiza de $L-c \leq x \leq e$), las rotaciones obtenidas de las dos ecuaciones son iguales.

4. Sustituyendo el valor de "$x = a$" en las ecuaciones (17) y (19) (Cuando la carga se localiza de $e \leq x \leq a$), en la ecuación (31) y (33) (Cuando la carga se localiza de $a \leq x \leq L-c$), en la ecuación (47) y (49) (Cuando la carga se localiza de $L-c \leq x \leq L$), los desplazamientos obtenidos de las dos ecuaciones son iguales.

5. Sustituyendo el valor de "$x = L-c$" en las ecuaciones (19) y (21) (Cuando la carga se localiza de $0 \leq x \leq a$), en la ecuación (35) y (37) (Cuando la carga se localiza de

ISSN 2007-9737





$e = 0.1L$ $e = 0.2L$ $e = 0.3L$

$e = 0.4L$ $e = 0.5L$ $e = 0.6L$

$e = 0.7L$ $e = 0.8L$ $e = 0.9L$

$h = 0.1LMP$ $h = 0.1LMA$ $h = 0.05LMP$ $h = 0.05LMA$

**Fig. 3.** Curva elástica en vigas simplemente apoyadas

$a \leq x \leq L - c$), en la ecuación (49) y (51) (Cuando la carga se localiza de $L - c \leq x \leq e$), los desplazamientos obtenidos de las dos ecuaciones son iguales.

6. Sustituyendo el valor de "$x = e$" en las ecuaciones (15) y (17) (Cuando la carga se localiza de $0 \leq x \leq a$), en la ecuación (33) y (35) (Cuando la carga se localiza de $a \leq x \leq L - c$), en la ecuación (51) y (53) (Cuando la carga se localiza de $L - c \leq x \leq e$), los desplazamientos obtenidos de las dos ecuaciones son iguales.

Una manera de verificar el modelo propuesto en algunos puntos especiales es como siguiente:

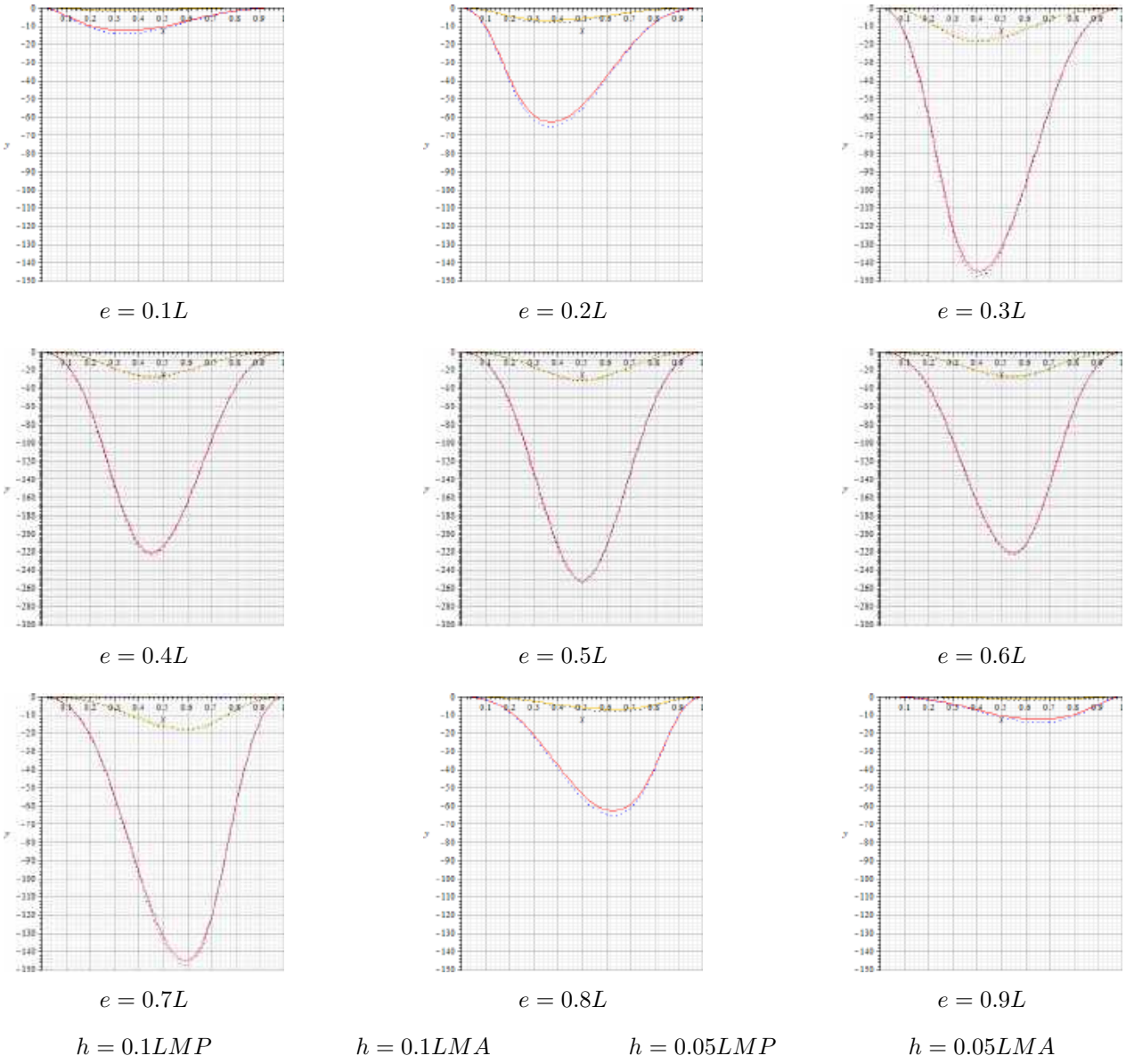
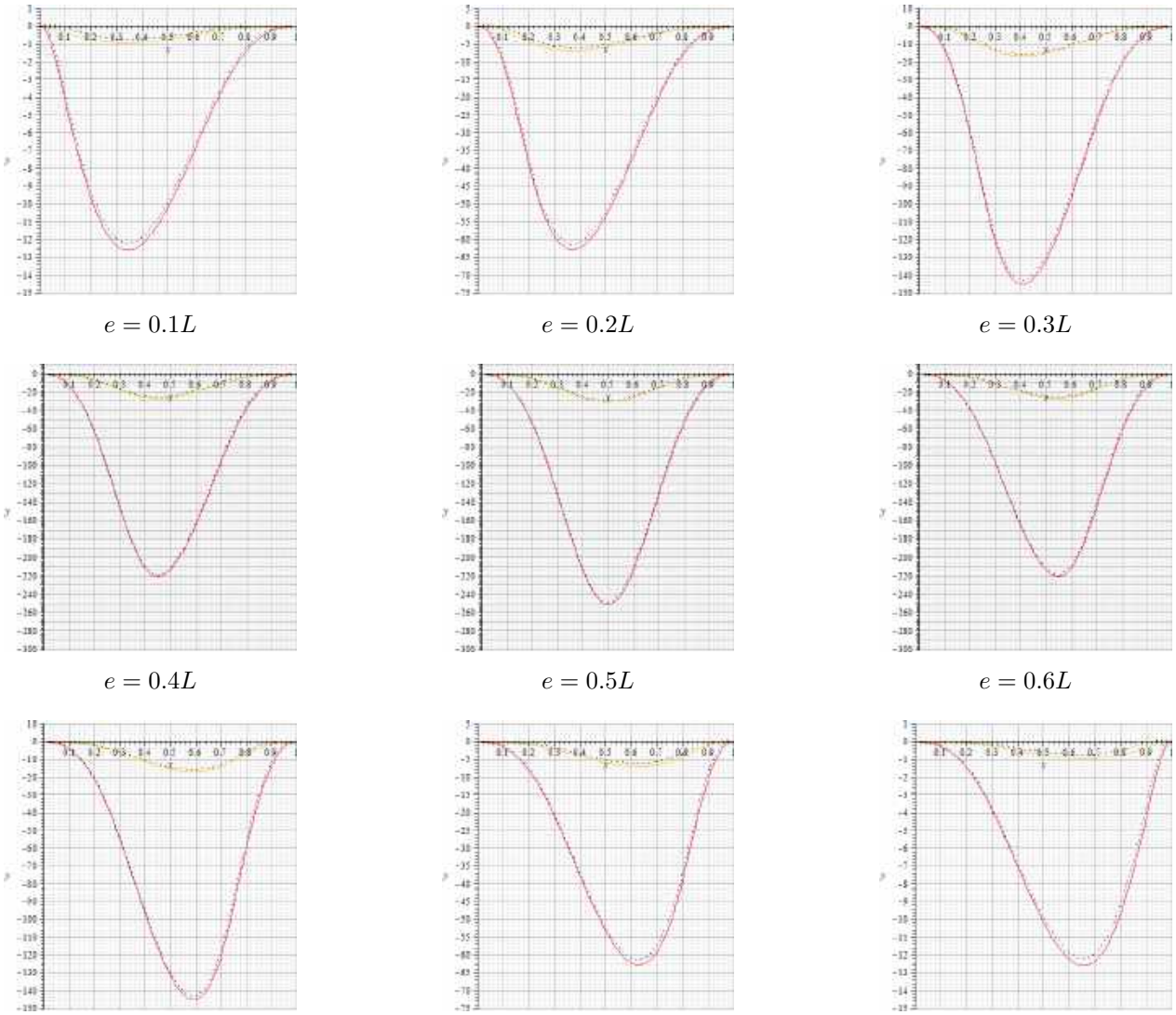



**Fig. 4.** Curva elástica en vigas perfectamente fijadas en sus extremos

1. Sustituyendo el valor de "$x = 0$" en las ecuaciones (15), (31) y (47) se obtiene el desplazamiento cero en el apoyo A.
2. Sustituyendo el valor de "$x = L$" en las ecuaciones (21), (37) y (53) se obtiene el desplazamiento cero en el apoyo B.
3. Sustituyendo el valor de $M_{AB} = 0$, $R_A = P/2$, $e = L/2$, $a = c$, $u = s$ y $dy/dx = 0$ para vigas de sección transversal rectangular simplemente apoyadas en las ecuaciones (32) y (34) se obtiene "$x = L/2$", es decir, cuando la rotación es cero, se produce el desplazamiento máximo (viga simétrica).



$e = 0.1L$ $e = 0.2L$ $e = 0.3L$

$e = 0.4L$ $e = 0.5L$ $e = 0.6L$

$e = 0.7L$ $e = 0.8L$ $e = 0.9L$

$h = 0.1LMP$ $h = 0.1LMA$ $h = 0.05LMP$ $h = 0.05LMA$

**Fig. 5.** Curva elástica en vigas usando los mismos momentos y reacciones en los extremos

La curva elástica para vigas rectangulares con cartelas parabólicas bajo carga concentrada situada en cualquier parte de la viga se obtiene, sustituyendo en las ecuaciones correspondientes de la siguiente manera:

1. Para vigas simplemente apoyadas:
   - $M_{AB} = 0$,
   - $R_{AB} = 0.9P(e = 0.1L),\ 0.8P(e = 0.2L),\ 0.7P(e = 0.3L),\ 0.6P(e = 0.4L),\ 0.5P(e = 0.5L),\ 0.4P(e = 0.6L),\ 0.3P(e = 0.7L),\ 0.2P(e = 0.8L),\ 0.1P(e = 0.9L)$,
   - $G = 5E/12$,

**Tabla 2.** Deflexiones en vigas simplemente apoyadas

| e | Factores de las rotaciones en el apoyo A y en el apoyo B | | | | | | Factores del desplazamiento máximo y su ubicación a partir del apoyo A | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\beta_{AB}$ | | | $\beta_{BA}$ | | | $\epsilon$ | | | $\rho$ | | |
| | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** |
| | | | | | $h = 0.1L,\ \ a = 0.3L,\ \ c = 0.3L$ | | | | | | | |
| 0.1L | 256.5180 | 252.3370 | 1.0166 | -174.3076 | -172.9501 | 1.0078 | 0.4398 | 0.4397 | 1.0002 | -68.3887 | -68.2495 | 1.0020 |
| 0.2L | 475.7573 | 467.5538 | 1.0175 | -344.4126 | -341.0962 | 1.0097 | 0.4429 | 0.4432 | 0.9993 | -134.4296 | -133.8157 | 1.0046 |
| 0.3L | 623.7549 | 614.4166 | 1.0152 | -496.1313 | -491.1697 | 1.0101 | 0.4532 | 0.4535 | 0.9993 | -190.3910 | -189.4294 | 1.0051 |
| 0.4L | 687.8740 | 679.6049 | 1.0122 | -612.2122 | -605.9814 | 1.0103 | 0.4724 | 0.4725 | 0.9998 | -227.2434 | -226.3062 | 1.0041 |
| 0.5L | 679.9931 | 672.7931 | 1.0107 | -679.9931 | -672.7931 | 1.0107 | 0.5000 | 0.5000 | 1.0000 | -240.2943 | -239.3673 | 1.0039 |
| 0.6L | 612.2122 | 605.9814 | 1.0103 | -687.8740 | -679.6049 | 1.0122 | 0.5276 | 0.5275 | 1.0002 | -227.2434 | -226.3062 | 1.0041 |
| 0.7L | 496.1313 | 491.1697 | 1.0101 | -623.7549 | -614.4166 | 1.0152 | 0.5468 | 0.5465 | 1.0005 | -190.3910 | -189.4294 | 1.0051 |
| 0.8L | 344.4126 | 341.0962 | 1.0097 | -475.7573 | -467.5538 | 1.0175 | 0.5571 | 0.5568 | 1.0005 | -134.4296 | -133.8157 | 1.0046 |
| 0.9L | 174.3076 | 172.9501 | 1.0078 | -256.5180 | -252.3370 | 1.0166 | 0.5602 | 0.5603 | 0.9998 | -68.3887 | -68.2495 | 1.0020 |
| | | | | | $h = 0.05L,\ \ a = 0.3L,\ \ c = 0.3L$ | | | | | | | |
| 0.1L | 2027.0581 | 2018.6962 | 1.0041 | -1386.3156 | -1383.6005 | 1.0020 | 0.4397 | 0.4397 | 1.0000 | -546.3744 | -545.9960 | 1.0007 |
| 0.2L | 3756.8373 | 3740.4302 | 1.0044 | -2735.4021 | -2728.7693 | 1.0024 | 0.4432 | 0.4432 | 1.0000 | -1071.7534 | -1070.5257 | 1.0011 |
| 0.3L | 4934.0093 | 4915.3326 | 1.0038 | -3939.1808 | -3929.3574 | 1.0025 | 0.4535 | 0.4535 | 1.0000 | -1517.3580 | -1515.4349 | 1.0013 |
| 0.4L | 5453.3772 | 5436.8388 | 1.0030 | -4860.4556 | -4847.8512 | 1.0026 | 0.4725 | 0.4725 | 1.0000 | -1812.3238 | -1810.4492 | 1.0010 |
| 0.5L | 5396.7450 | 5382.3450 | 1.0027 | -5396.7450 | -5382.3450 | 1.0027 | 0.5000 | 0.5000 | 1.0000 | -1916.7922 | -1914.9380 | 1.0010 |
| 0.6L | 4860.4556 | 4847.8512 | 1.0026 | -5453.3772 | -5436.8388 | 1.0030 | 0.5275 | 0.5275 | 1.0000 | -1812.3238 | 1810.4492 | 1.0010 |
| 0.7L | 3939.1808 | 3929.3574 | 1.0025 | -4934.0093 | -4915.3326 | 1.0038 | 0.5465 | 0.5465 | 1.0000 | -1517.3580 | -1515.4349 | 1.0013 |
| 0.8L | 2735.4021 | 2728.7693 | 1.0024 | -3756.8373 | -3740.4302 | 1.0044 | 0.5568 | 0.5568 | 1.0000 | -1071.7534 | -1070.5257 | 1.0011 |
| 0.9L | 1386.3156 | 1383.6005 | 1.0020 | -2027.0581 | -2018.6962 | 1.0041 | 0.5603 | 0.5603 | 1.0000 | -546.2744 | -545.9960 | 1.0007 |

donde: $\theta_{AB}$ (rotación en el apoyo $A$) $= \beta_{AB} P/Eb$; $\theta_{BA}$ (rotación en el apoyo $B$) $= \beta_{BA} P/Eb$; $x_{AB}$ (ubicación del momento máximo a partir del apoyo $A$) $= \epsilon L$; $y_{\text{max}}$ (desplazamiento máximo) $= \rho P/Eb$

- $a = 0.3L$,
- $u = h$,
- $c = 0.3L$,
- $s = h$,
- $h = 0.05L, 0.1L$.

Para los dos modelos, el MP (Modelo propuesto considera las deformaciones por flexión y cortante) y el MA (Modelo actual considera las deformaciones por flexión).

2. Para vigas perfectamente fijadas en sus extremos: El momento $M_{AB}$ y la reacción $R_A$ en el apoyo "$A$" se obtienen sustituyendo en las ecuaciones correspondientes:
   - $dy/dx = 0$,
   - $x = 0, L$,
   - $G = 5E/12$,
   - $a = 0.3L$,
   - $u = h$,
   - $c = 0.3L$,
   - $s = h$,
   - $e = 0.1L,\ 0.2L,\ 0.3L,\ 0.4L,\ 0.5L,\ 0.6L,\ 0.7L,\ 0.8L,\ 0.9L$
   - $h = 0.05L, 0.1L$.

   De los dos modelos (MP y MA), estas dos ecuaciones se resuelven para obtener "$M_{AB} = m_{AB} PL$" y la reacción "$R_A = r_{AB} P$" en el apoyo $A$. Posteriormente, sustituyendo los valores de $M_{AB}$ y $R_A$ para los dos modelos:
   - $G = 5E/12$,

**Tabla 3.** Deflexiones en vigas perfectamente fijadas en sus extremos

| e | Factores para momentos y reacciones en el apoyo A | | | | | | Factores del desplazamiento máximo y su ubicación a partir del apoyo A | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $m_{AB}$ | | | $\alpha_{AB}$ | | | $\epsilon$ | | | $\rho$ | | |
| | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** |
| $h = 0.1L, \quad a = 0.3L, \quad c = 0.3L$ | | | | | | | | | | | | |
| 0.1L | 0.0966 | 0.0932 | 1.0365 | 0.9943 | 0.9889 | 1.0055 | 0.3383 | 0.3452 | 0.9800 | -0.9263 | -1.7891 | 0.5177 |
| 0.2L | 0.1694 | 0.1635 | 1.0361 | 0.9507 | 0.9416 | 1.0097 | 0.3687 | 0.3714 | 0.9927 | -6.8415 | -8.1793 | 0.8364 |
| 0.3L | 0.2016 | 0.1958 | 1.0296 | 0.8463 | 0.8380 | 1.0099 | 0.4071 | 0.4079 | 0.9980 | -17.0884 | -18.4643 | 0.9255 |
| 0.4L | 0.1925 | 0.1886 | 1.0207 | 0.6869 | 0.6825 | 1.0064 | 0.4507 | 0.4507 | 1.0000 | -26.5992 | -27.9204 | 0.9527 |
| 0.5L | 0.1560 | 0.1543 | 1.0110 | 0.5000 | 0.5000 | 1.0000 | 0.5000 | 0.5000 | 1.0000 | -30.4482 | -31.7431 | 0.9592 |
| 0.6L | 0.1056 | 0.1062 | 0.9944 | 0.3131 | 0.3175 | 0.9861 | 0.5493 | 0.5493 | 1.0000 | -26.5992 | -27.9204 | 0.9527 |
| 0.7L | 0.0553 | 0.0578 | 0.9567 | 0.1537 | 0.1620 | 0.9488 | 0.5929 | 0.5921 | 1.0014 | -17.0884 | -18.4643 | 0.9255 |
| 0.8L | 0.0187 | 0.0219 | 0.8539 | 0.0493 | 0.0584 | 0.8442 | 0.6313 | 0.6286 | 1.0043 | -6.8415 | -8.1793 | 0.8364 |
| 0.9L | 0.0022 | 0.0043 | 0.5116 | 0.0057 | 0.0111 | 0.5135 | 0.6617 | 0.6548 | 1.0105 | -0.9263 | -1.7191 | 0.5177 |
| $h = 0.05L, \quad a = 0.3L, \quad c = 0.3L$ | | | | | | | | | | | | |
| 0.1L | 0.0940 | 0.0932 | 1.0086 | 0.9902 | 0.9889 | 1.0013 | 0.3441 | 0.3452 | 0.9968 | -12.5921 | -14.3128 | 0.8798 |
| 0.2L | 0.1650 | 0.1635 | 1.0092 | 0.9439 | 0.9416 | 1.0024 | 0.3708 | 0.3714 | 0.9984 | -62.7547 | -65.4342 | 0.9591 |
| 0.3L | 0.1972 | 0.1958 | 1.0072 | 0.8401 | 0.8380 | 1.0025 | 0.4077 | 0.4079 | 0.9995 | -144.9686 | -147.7146 | 0.9814 |
| 0.4L | 0.1896 | 0.1886 | 1.0053 | 0.6836 | 0.6825 | 1.0016 | 0.4507 | 0.4507 | 1.0000 | -220.7226 | -223.3631 | 0.9882 |
| 0.5L | 0.1547 | 0.1543 | 1.0026 | 0.5000 | 0.5000 | 1.0000 | 0.5000 | 0.5000 | 1.0000 | -251.3548 | -253.9444 | 0.9898 |
| 0.6L | 0.1061 | 0.1062 | 0.9991 | 0.3164 | 0.3175 | 0.9965 | 0.5493 | 0.5493 | 1.0000 | -220.7226 | -223.3631 | 0.9882 |
| 0.7L | 0.0572 | 0.0578 | 0.9896 | 0.1599 | 0.1620 | 0.9870 | 0.5923 | 0.5921 | 1.0003 | -144.9686 | -147.7146 | 0.9814 |
| 0.8L | 0.0211 | 0.0219 | 0.9635 | 0.0561 | 0.0584 | 0.9606 | 0.6292 | 0.6286 | 1.0009 | -62.7547 | -65.4342 | 0.9591 |
| 0.9L | 0.0038 | 0.0043 | 0.8837 | 0.0098 | 0.0111 | 0.8829 | 0.6559 | 0.6548 | 1.0017 | -12.5921 | -14.3128 | 0.8798 |

dónde: $M_{AB}$ (momento de empotramiento en el apoyo A) $= m_{AB}PL$; $R_A$ (reacción en el apoyo A) = $\alpha_{AB}P$

- $a = 0.3L$,
- $u = h$,
- $c = 0.3L$,
- $s = h$,
- $e = 0.1L,\ 0.2L,\ 0.3L,\ 0.4L,\ 0.5L,\ 0.6L,\ 0.7L,\ 0.8L,\ 0.9L$
- $h = 0.05L, 0.1L$.

3. Para vigas usando los mismos momentos y reacciones son del modelo propuesto para los dos modelos.

La Figura 3 muestra la curva elástica para vigas rectangulares simplemente apoyadas con cartelas parabólicas bajo carga concentrada situada en cualquier parte de la viga para los dos modelos (MP y MA).

La Figura 4 muestra la curva elástica para vigas perfectamente fijadas en sus extremos con cartelas parabólicas bajo carga concentrada situada en cualquier parte de la viga para los dos modelos (MP y MA).

La Figura 5 muestra la curva elástica para vigas usando los mismos momentos y reacciones en sus extremos del modelo propuesto con cartelas parabólicas bajo carga concentrada situada en cualquier parte de la viga para los dos modelos (MP y MA).

## 4. Resultados

Los valores de la curva elástica (deflexiones) para la viga simplemente apoyada es mayor para el modelo propuesto con respecto al modelo actual para $h = 0.1L$ y $h = 0.05L$, y los valores mayores se presentan para $h = 0.05L$ (ver Figura 3).

**Tabla 4.** Deflexiones en vigas usando los momentos del modelo propuesto para los dos modelos

| e | Factores de las rotaciones en el apoyo A y en el apoyo B | | | | Factores del desplazamiento máximo y su ubicación a partir del apoyo A | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $\beta_{AB}$ | | $\beta_{BA}$ | | $\epsilon$ | | | $\rho$ | | |
| | **MP** | **MA** | **MP** | **MA** | **MP** | **MA** | **MP/MA** | **MP** | **MA** | **MP/MA** |
| $h = 0.1L, \quad a = 0.3L, \quad c = 0.3L$ | | | | | | | | | | |
| 0.1L | 0 | -5.1895 | 0 | 0.3489 | 0.3383 | 0.3535 | 0.9570 | -0.9263 | -0.7409 | 1.2502 |
| 0.2L | 0 | -9.8149 | 0 | 1.7051 | 0.3687 | 0.3754 | 0.9822 | -6.8415 | -6.1251 | 1.1170 |
| 0.3L | 0 | -10.9028 | 0 | 3.4972 | 0.4071 | 0.4101 | 0.9927 | -17.0884 | -16.0440 | 1.0651 |
| 0.4L | 0 | -9.1986 | 0 | 5.2014 | 0.4507 | 0.4516 | 0.9980 | -26.5992 | -25.6383 | 1.0375 |
| 0.5L | 0 | -7.2000 | 0 | 7.2000 | 0.5000 | 0.5000 | 1.0000 | -30.4482 | -29.5211 | 1.0314 |
| 0.6L | 0 | -5.2014 | 0 | 9.1986 | 0.5493 | 0.5484 | 1.0016 | -26.5992 | -25.6383 | 1.0375 |
| 0.7L | 0 | -3.4972 | 0 | 10.9028 | 0.5929 | 0.5899 | 1.0051 | -17.0884 | -16.0440 | 1.0651 |
| 0.8L | 0 | -1.7051 | 0 | 9.8149 | 0.6313 | 0.6246 | 1.0107 | -6.8415 | -6.1251 | 1.1170 |
| 0.9L | 0 | -0.3489 | 0 | 5.1895 | 0.6617 | 0.6465 | 1.0235 | -0.9263 | -0.7409 | 1.2502 |
| $h = 0.05L, \quad a = 0.3L, \quad c = 0.3L$ | | | | | | | | | | |
| 0.1L | 0 | -10.2916 | 0 | 0.7854 | 0.3441 | 0.3461 | 0.9942 | -12.5921 | -12.2257 | 1.0300 |
| 0.2L | 0 | -19.4833 | 0 | 3.5567 | 0.3708 | 0.3722 | 0.9962 | -62.7547 | -61.3273 | 1.0233 |
| 0.3L | 0 | -21.6718 | 0 | 7.1282 | 0.4077 | 0.4084 | 0.9983 | -144.9686 | -142.8824 | 1.0146 |
| 0.4L | 0 | -18.3250 | 0 | 10.4750 | 0.4507 | 0.4509 | 0.9996 | -220.7226 | -218.8014 | 1.0088 |
| 0.5L | 0 | -14.4000 | 0 | 14.4000 | 0.5000 | 0.5000 | 1.0000 | -251.3548 | -249.5006 | 1.0074 |
| 0.6L | 0 | -10.4750 | 0 | 18.3250 | 0.5493 | 0.5491 | 1.0004 | -220.7226 | -218.8014 | 1.0088 |
| 0.7L | 0 | -7.1282 | 0 | 21.6718 | 0.5923 | 0.5916 | 1.0012 | -144.9686 | -142.8824 | 1.0146 |
| 0.8L | 0 | -3,5567 | 0 | 19.4833 | 0.6292 | 0.6278 | 1.0022 | -62.7547 | -62.3273 | 1.0233 |
| 0.9L | 0 | -0.7854 | 0 | 10.2916 | 0.6559 | 0.6539 | 1.0031 | -12.5921 | -12.2257 | 1.0300 |

Para la viga perfectamente fijada en sus extremos es mayor para el modelo actual con respecto al modelo propuesto para $h = 0.1L$ y un valor de $h = 0.05L$, y los valores mayores son para $h = 0.05L$ (ver Figura 4).

Para la viga usando los momentos del modelo propuesto para los dos modelos es mayor para el modelo propuesto con respecto al modelo actual para $h = 0.1L$ y $h = 0.05L$, y los valores mayores son para $h = 0.05L$ (ver Figura 5).

La Tabla 2 muestra la comparación de los dos modelos para obtener los factores para las rotaciones en los apoyos y los desplazamientos máximos para una viga simplemente apoyada.

Estas comparaciones se realizan para $G = 5E$; $h = 0.1L$ además de $h = 0.05L$. La Tabla 2 muestra lo siguiente:

1. Las rotaciones y desplazamientos son mayores en el modelo propuesto con respecto al modelo actual.

2. Las rotaciones mayores aparecen cuando la carga $P$ se ubica en $e = 0.4L$ para $\theta_{AB}$, y cuando la carga $P$ se ubica en $e = 0.6L$ para $\theta_{BA}$ para los dos modelos ($h = 0.1L$ y $h = 0.05L$).

3. La diferencia mayor de las rotaciones para los dos modelos aparecen en $e = 0.2L$ en $\theta_{BA}$ de $1.0175$ y $e = 0.8L$ en $\theta_{BA}$ de $1.0175$ para

$h = 0.1L$, y en $e = 0.2L$ en $\theta_{BA}$ de $1.0044$ y $e = 0.8L$ en $\theta_{BA}$ de $1.0044$ para $h = 0.05L$.

4. Los desplazamientos máximos aparecen cuando la carga $P$ se ubica en $e = 0.5L$ para los dos modelos ($h = 0.1L$ y $h = 0.05L$).
5. La diferencia mayor de los desplazamientos máximos para los dos modelos aparecen en $e = 0.3L$ y $e = 0.7L$. Para $h = 0.1L$ es de $1.0051$ y para $h = 0.05L$ es de $1.0013$.

La Tabla 3 muestra la comparación de los dos modelos para obtener los factores para las rotaciones en los apoyos y los desplazamientos máximos para una viga empotrada en ambos extremos. Estas comparaciones se realizan para $G = 5E$; $h = 0.1L$ y $h = 0.05L$. La Tabla 3 muestra lo siguiente:

1. Los momentos $M_{AB}$ y las reacciones $R_A$ en el modelo propuesto son mayores desde $e = 0.1L$ hasta $e = 0.5L$ y son menores desde $e = 0.6L$ hasta $e = 0.9L$ con respecto al modelo actual.
2. Los momentos $M_{AB}$ mayores aparecen cuando la carga $P$ se ubica en $e = 0.3L$ para los dos modelos con $h = 0.1L$ y $h = 0.05L$.
3. Las reacciones $R_A$ mayores aparecen cuando la carga $P$ se ubica en $e = 0.1L$ para los dos modelos con $h = 0.1L$ y $h = 0.05L$.
4. Los desplazamientos máximos en el modelo actual son mayores con respecto al modelo propuesto.
5. Los desplazamientos máximos aparecen cuando la carga $P$ se ubica en $e = 0.5L$ para los dos modelos ($h = 0.1L$ y $h = 0.05L$).
6. La mayor diferencia de los desplazamientos máximos entre los dos modelos aparece en $e = 0.1L$ y $e = 0.9L$. Para $h = 0.1L$ es de $0.5177$ y para $h = 0.05L$ es de $0.8798$.

La Tabla 4 muestra la comparación de los dos modelos para obtener los factores para las rotaciones en los apoyos y los desplazamientos máximos para una viga usando los momentos del modelo propuesto para los dos modelos. Estas comparaciones se realizan para $G = 5E$; $h = 0.1L$ además de $h = 0.05L$. Los resultados mostrados en la Tabla 4 son los siguientes:

1. Las rotaciones en el modelo propuesto son cero, porque los momentos son para vigas perfectamente fijadas en sus extremos.
2. Las rotaciones máximas aparecen en $e = 0.3L$ para el apoyo $A$ y en $e = 0.7L$ para el apoyo $B$ en el modelo actual ($h = 0.1L$ y $h = 0.05L$).
3. Los desplazamientos máximos en el modelo propuesto son mayores con respecto al modelo actual.
4. Los desplazamientos máximos aparecen cuando la carga $P$ se ubica en $e = 0.5L$ para los dos modelos ($h = 0.1L$ y $h = 0.05L$).
5. La mayor diferencia de los desplazamientos máximos entre los dos modelos aparece en $e = 0.1L$ y $e = 0.9L$. Para $h = 0.1L$ es de $1.2502$ y para $h = 0.05L$ es de $1.0300$.

## 5. Conclusiones

El modelo presentado en este documento para una viga de sección transversal rectangular con cartelas parabólicas bajo una carga concentrada localizada en cualquier parte de la viga y momentos en los extremos para obtener las rotaciones, los desplazamientos y la curva elástica, considerando las deformaciones por flexión y cortante (teoría de Timoshenko) ha sido desarrollado para el caso general. Las principales conclusiones son:

1. En el modelo actual la relación "$h/L$" no influye:
   a) Para vigas simplemente apoyadas, la posición de la carga.
   b) Para vigas empotradas en sus extremos, los momentos, las reacciones en los apoyos y la posición de la carga.
2. El modelo propuesto es mayor con respecto al modelo actual para las rotaciones en los apoyos y los desplazamientos máximos, cuando se presentan los mismos momentos en los extremos.

3. El modelo propuesto supera al actual en el intervalo de $e = 0.1L$ a $e = 0.5L$, mientras que es inferior entre $e = 0.6L$ y $e = 0.9L$ para los momentos $M_{AB}$ y las reacciones $R_A$, cuando la viga está perfectamente fijada en ambos extremos.

4. El modelo propuesto es menor con respecto al modelo actual para los desplazamientos máximos, cuando la viga está perfectamente fijada en sus extremos.

Los desplazamientos máximos por el modelo propuesto que considera las deformaciones por flexión y cortante son mayores para vigas con los mismos momentos en los extremos, y son menores para vigas perfectamente fijadas en sus extremos respecto al modelo actual que considera las deformaciones por flexión.

Por lo tanto, las deflexiones máximas que actúan sobre las vigas del modelo propuesto en este trabajo deben compararse con las deflexiones máximas permitidas por los códigos de construcción, porque en algunas condiciones podría ser que no cumpla con los estándares establecidos por los códigos de construcción.

Entonces, el modelo propuesto es más apropiado y seguro con respecto al modelo actual para el análisis estructural, debido a que los esfuerzos cortantes y los momentos flectores están presentes en cualquier tipo de estructura y aparecen deformaciones por flexión y cortante. Las sugerencias para investigaciones futuras pueden ser:

1. Cuando la sección transversal de la viga sea una `I` o `T`.

2. Cuando la carga aplicada es diferente a una carga concentrada o carga uniformemente distribuida.

## 6. Agradecimientos

La investigación descrita en este trabajo fue financiada por el Instituto de Investigaciones Multidisciplinarias de la Facultad de Contaduría y Administración de la Universidad Autónoma de Coahuila. Los autores también agradecen a los revisores y al editor por los comentarios y sugerencias para mejorar la presentación. El estudiante de doctorado Rogelio Barraza Saucedo (CVU: 1013684) agradece al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo económico.

## Referencias

1. **Banerjee, A., Bhattacharya, B., Mallik, A. (2008).** Large deflection of cantilever beams with geometric non-linearity: Analytical and numerical approaches. International Journal of Non-Linear Mechanics, Vol. 43, No. 5, pp. 366–376. DOI: 10.1016/j.ijnonlinmec.2007.12.020.

2. **Borboni, A., De-Santis, D. (2014).** Large deflection of a non-linear, elastic, asymmetric ludwick cantilever beam subjected to horizontal force, vertical force and bending torque at the free end. Meccanica, Vol. 49, No. 6. DOI: 10.1007/s11012-014-9895-z.

3. **Brojan, M., Cebron, M., Kosel, F. (2012).** Large deflections of non-prismatic nonlinearly elastic cantilever beams subjected to non-uniform continuous load and a concentrated load at the free end. Acta Mechanica Sinica, Vol. 28, No. 3, pp. 863–869. DOI: 10.1007/s10409-012-0053-3.

4. **Chen, L. (2010).** An integral approach for large deflection cantilever beams. International Journal of Non-Linear Mechanics, Vol. 45, No. 3, pp. 301–305. DOI: 10.1016/j.ijnonlinmec.2009.12.004.

5. **Crispín-Herrera, C. Y., Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M. (2021).** Design aids for beams of rectangular cross section with parabolic haunches: Part 1. Computación y Sistemas, Vol. 25, No. 3, pp. 633-646. DOI: 10.13053/cys-25-3-3777.

6. **Dado, M., Al-Sadder, S. (2005).** A new technique for large deflection analysis of non-prismatic cantilever beams. Mechanics Research Communications, Vol. 32, No. 6, pp. 692–703. DOI: 10.1016/j.mechrescom.2005.01.004.

7. **Debnath, V., Debnath, B. (2014).** Deflection and stress analysis of a beam on different elements using ANSYS APDL. International Journal of Mechanical Engineering and Technology, Vol. 5, No. 6, pp. 70-79.

8. **Gaona-Tamez, L. L., Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M., Jaramillo-Rosales, M. (2020).** Design aids for rectangular cross-section beams with straight haunches: Part 2. International Journal of Innovative Computing, Information and Control, Vol. 16, No. 6, pp. 1929–1942. DOI: 10.24507/ijicic.16.06.1929.

9. **Lee, K. (2002).** Large deflections of cantilever beams of non-linear elastic material under a combined loading. International Journal of Non-Linear Mechanics, Vol. 37, No. 3, pp. 439–443. DOI: 10.1016/s0020-7462(01) 00019-1.

10. **Luévanos-Rojas, A., García-Canales, E., López-Chavarría, S., Medina-Elizondo, M. (2020).** Minimum cost for reinforced concrete rectangular beams with parabolic haunches. Computación y Sistemas, Vol. 24, No. 3, pp. 1063–1073. DOI: 10.13053/cys-24-3-3306.

11. **Luévanos-Rojas, A., Luévanos-Soto, R. M., López-Chavarría, S., Medina-Elizondo, M. (2021).** Design aids for beams of rectangular cross section with parabolic haunches: Part 2. Computación y Sistemas, Vol. 25, No. 4, pp. 821–834. DOI: 10.13053/cys-25-4-3778.

12. **Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M. (2016).** Modeling for mechanical elements of rectangular members with straight haunches using software: Part 1. International Journal of Innovative Computing, Information and Control, Vol. 12, No. 3, pp. 973–985. DOI: 10.24507/ijicic.12.03.959.

13. **Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M. (2016).** Modeling for mechanical elements of rectangular members with straight haunches using software: Part 2. International Journal of Innovative Computing, Information and Control, Vol. 12, No. 4, pp. 1027–1041. DOI: 10.24507/ijicic.12.04. 1027.

14. **Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M., Kalashnikov, V. (2014).** A mathematical model of elastic curve for simply supported beams subjected to a uniformly distributed load taking into account the shear deformations. ICIC Express Letters Part B: Applications, Vol. 5, No. 3, pp. 885–890.

15. **Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M., Kalashnikov, V. V. (2020).** Optimal design of reinforced concrete beams for rectangular sections with straight haunches. Revista de la construcción, Vol. 19, No. 1, pp. 90–102. DOI: 10.7764/rdlc.19.1. 90-102.

16. **Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M., Kalashnikov, V. V. (2016).** A mathematical model of elastic curve for simply supported beams subjected to a concentrated load taking into account the shear deformations. International Journal of Innovative Computing, Information and Control, Vol. 12, No. 1, pp. 41–54. DOI: 10. 24507/ijicic.12.01.41.

17. **Majumder, G., Kumar, K. (2013).** Deflection and stress analysis of a simply supported beam and its validation using ANSYS. International Journal of Mechanical Engineering and Computer Applications, Vol. 1, No. 5, pp. 17–20.

18. **Montano-Perez, B. E., Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M., Jaramillo-Rosales, M. (2020)**. Design aids for rectangular cross-section beams with straight haunches: Part 1. International Journal of Innovative Computing, Information and Control, Vol. 16, No. 6, pp. 1915–1928. DOI: 10.24507/ijicic.16.06.1915.

19. **Sandoval-Rivas, R., Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M. (2019)** Modelado para trabes de sección transversal rectangular con cartelas parabólicas: parte 2. Computación y

Sistemas, Vol. 23, No. 3, pp. 1115–1124. DOI: 10.13053/cys-23-3-2873.

20. **Sihua, D., Ze, Q., Li, W. (2015).** Nonlinear analysis of reinforced concrete beam bending failure experimentation based on abaqus. International Conference on Information Sciences, Machinery, Materials and Energy, pp. 440–444. DOI: 10.2991/icismme-15.2015.88.
21. **Solano-Carrillo, E. (2009).** Semi-exact solutions for large deflections of cantilever beams of non-linear elastic behaviour. International Journal of Non-Linear Mechanics, Vol. 44, No. 2, pp. 253–256. DOI: 10.1016/j.ijnonlinmec.2008.11.007.
22. **Timoshenko, S. P. (1947).** Strength of Materials. Part I: Elementary Theory and Problems, 2nd Edition, Van Nostrand Company, Inc., New York.
23. **Timoshenko, S. P., Gere, J. M. (1972).** Mechanics of materials. Van Nostrand Reinhold, New York.
24. **Velázquez-Santillán, F., Luévanos-Rojas, A., López-Chavarría, S., Medina-Elizondo, M. (2019).** Modeling for beams of rectangular cross section with parabolic haunches: Part 1. Computación y Sistemas, Vol. 23, No. 2, pp. 557–568. DOI: 10.13053/cys-23-2-2872.
25. **Yau, J. (2010).** Closed-form solutions of large deflection for a guyed cantilever column pulled by an inclination cable. Journal of Marine Science and Technology, Vol. 18, No. 1. DOI: 10.51400/2709-6998.1874.

*Article received on 10/01/2023; accepted on 08/04/2024.*
**Corresponding author is Arnulfo Luévanos Rojas.*

ISSN 2007-9737

# Performance Comparison of Stereo Correspondence Algorithms in Dense Image Matching

Seyyid Ahmed-Medjahed[1,*], Fatima Boukhatem[2]

[1] University of Relizane,
Algeria

[2] University of Djillali Liabes, Sidi Bel Abbes,
Algeria

seyyidahmed.medjahed@univ-relizane.dz,
fatima.boukhatem@univ-sba.dz

**Abstract.** Stereo matching is one of the most active research fields in computer vision. The aim of stereo matching is to find the corresponding points in two or more images that correspond to the same physical entity in the scene. In this paper, we deal the problem of stereo vision and precisely the dense stereo matching of images using correlation measures and other algorithms. We consider eight correlation techniques, subpixel estimation, dynamic programming and Hierarchical matching. The aim of this evaluation is to show the performance of each method in dense stereo matching images and in the 3D reconstruction. We also consider noisy image pairs with different noise level. The performance evaluation of each technique is conducted in term of: computational time, mean absolute error, mean relative error and the percentage of correct matches as well as wrong matches. A 3D reconstruction will be considered for the better methods.

**Keywords.** Computer vision, stereo matching, correlation, correspondence.

## 1 Introduction

Computer vision is a process of information processing that allows the machine to understand the reality by analyzing and interpreting the information.

It is a branch of artificial intelligence whose goal is to allow a machine understands what it “sees” when it is connected to one or more cameras. Stereo matching is one of the most difficult problems in computer vision. It is a fundamental low-level vision task for many applications such as autonomous vehicle navigation, objet manipulation, robot vision and biometrics [1, 11, 6]. Indeed, stereo matching is the fundamental precursor of 3D reconstruction from a stereo pair or sequence of images.

It consists to find the corresponding points in two or more images that correspond to the same physical entity in the scene. The result of image matching is the disparity map; this map contains the spatial shift for each pixel, but, it does not allow the knowledge of the 3d structure of the observed scene [7].

Local methods are based on an analysis of the neighborhood of the points to be matched. Several methods have been developed in this context, the most used are: SAD, SSD, ZNCC, etc. These methods are very effective when it comes to doing 3D reconstruction in images with low motion and a small variation of the camera’s intrinsic parameters.

The advantage of these methods lies in the fact that they require few resources and they generate images with a dense disparity. But, they have a high rate of error especially in the occlusion zones and in the texture zones.

**Table 1.** Some correlation methods

| Methods | Formulation |
|---|---|
| **Cross Correlation** | |
| NCC | $\mathrm{NCC}(w_l, w_r) = \frac{w_l \cdot w_r}{\|w_l\| \, \|w_r\|}$ |
| ZNCC | $\mathrm{ZNCC}(w_l, w_r) = \frac{(w_l - \overline{w_l}) \cdot (w_r - \overline{w_r})}{\|w_l - \overline{w_l}\| \, \|w_r - \overline{w_r}\|}$ |
| **Distance and Locally Centered Distance** | |
| SSD | $\mathrm{SSD}(w_l, w_r) = \|w_l - \overline{w_l}\|^2$ |
| SAD | $\mathrm{SAD}(w_l, w_r) = \|w_l - \overline{w_l}\|$ |
| ZSSD | $\mathrm{ZSSD}(w_l, w_r) = \|w_l - \overline{w_l}\| - \|w_r - \overline{w_r}\|^2$ |
| ZSAD | $\mathrm{ZSAD}(w_l, w_r) = \|w_l - \overline{w_l}\| - \|w_r - \overline{w_r}\|$ |
| LSSD | $\mathrm{LSSD}(w_l, w_r) = \left\| w_l - \frac{\overline{w_l}}{\overline{w_r}} w_r \right\|^2$ |
| LSAD | $\mathrm{LSAD}(w_l, w_r) = \left\| w_l - \frac{\overline{w_l}}{\overline{w_r}} w_r \right\|$ |

Other types of approaches are used for matching, these approaches are grouped together as global methods. The use of this type of method always leads us to a problem of minimizing an energy function defined over the entire image.

In matching, one is placed in the case of a combinatorial optimization. In this paper, we demonstrate the performances of several stereo matching algorithms: correlation techniques, subpixel estimation, dynamic programming and Hierarchical matching.

The experimental results is conducted on image pair obtained from Middlebury data base with different noise level. We consider the mean absolute error, mean relative error, the percentage of correct matches and wrong matches, computational time, as the performance measures.

Finally, we want to assess the impact of noise on these algorithms as well as algorithms parameters such as the correlation window size and the correlation function. The rest of paper is organized as follows:

In the next section, an overview of the different stereo matching algorithms is given. In section $3$, we present the evaluation criteria. In section $4$, we discuss the results obtained by the different approaches and finally we conclude by some perspectives.

## 2 Matching Algorithms

In general, the matching consists in finding in the left and right images, the homologous primitives, that is to say, the primitives that are the projection of the same entity of the scene. There are two types of matching: Dense matching: for each point of an image we calculate its correspondent.

Scattered matching: the correspondent is calculated only for points describing particular properties (the points of contours, corner, etc ...). The results of the mapping are visually represented by an image called the disparity map.

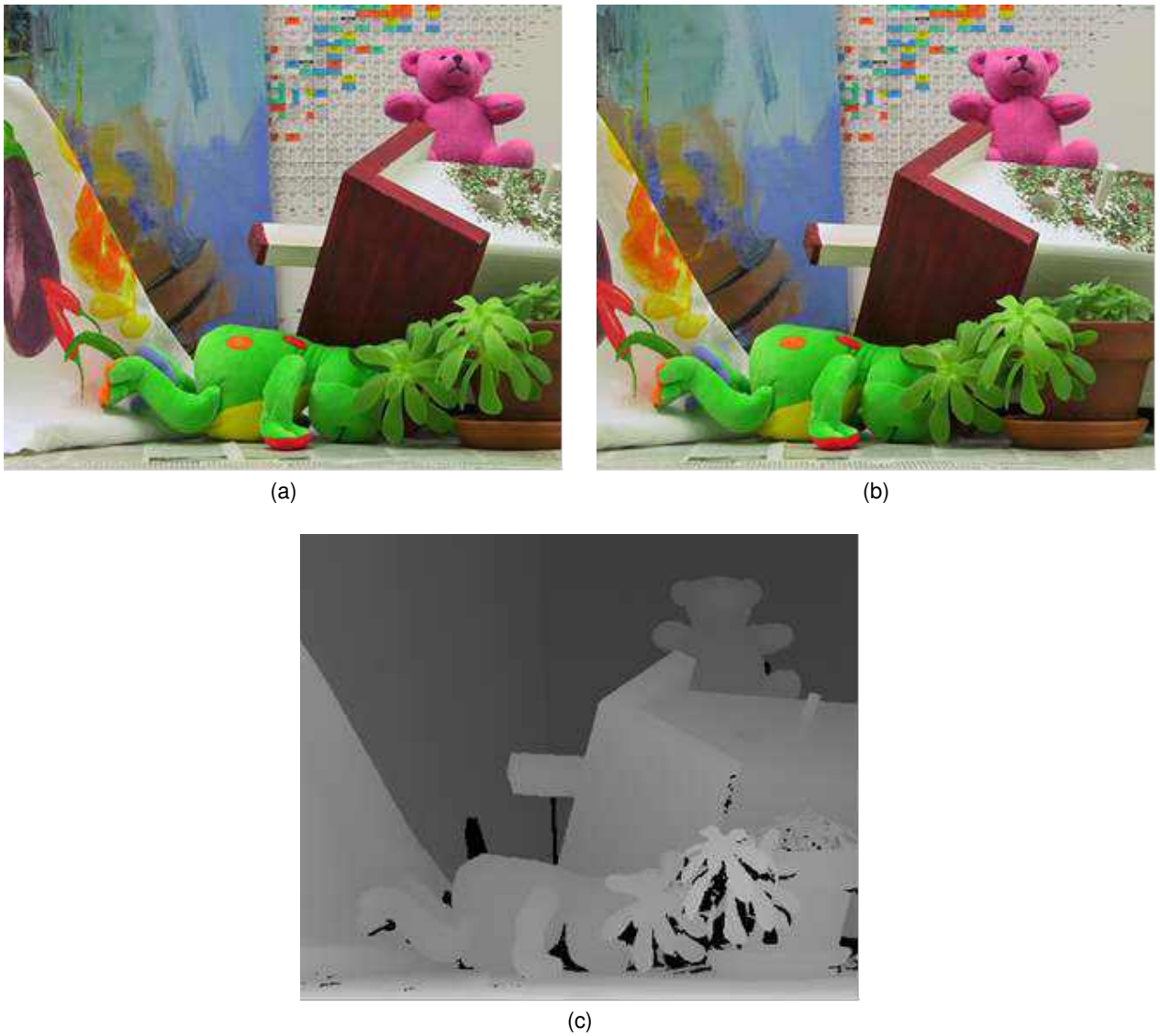
Each pixel of the disparity map represents the distance between the pixel position of the left image and that of its corresponding in the right image. In this study, we consider the following stereo matching algorithms:

### 2.1 Dynamic Programming

Introduced by Bellman and Dreyfus, dynamic programming has an important role in optimization problem. In computer vision, the Viterbi algorithm is used to estimate the disparity map between two images. The purpose of this technique is to find the optimal path from one side of the image to the other by using the block matching metric as the cost function [13, 8]. Many researchers have been interested by improve the dynamic programming algorithm and defined an iterated algorithm [4], also, parallel algorithm [9, 2].

ISSN 2007-9737





(a) (b)

(c)

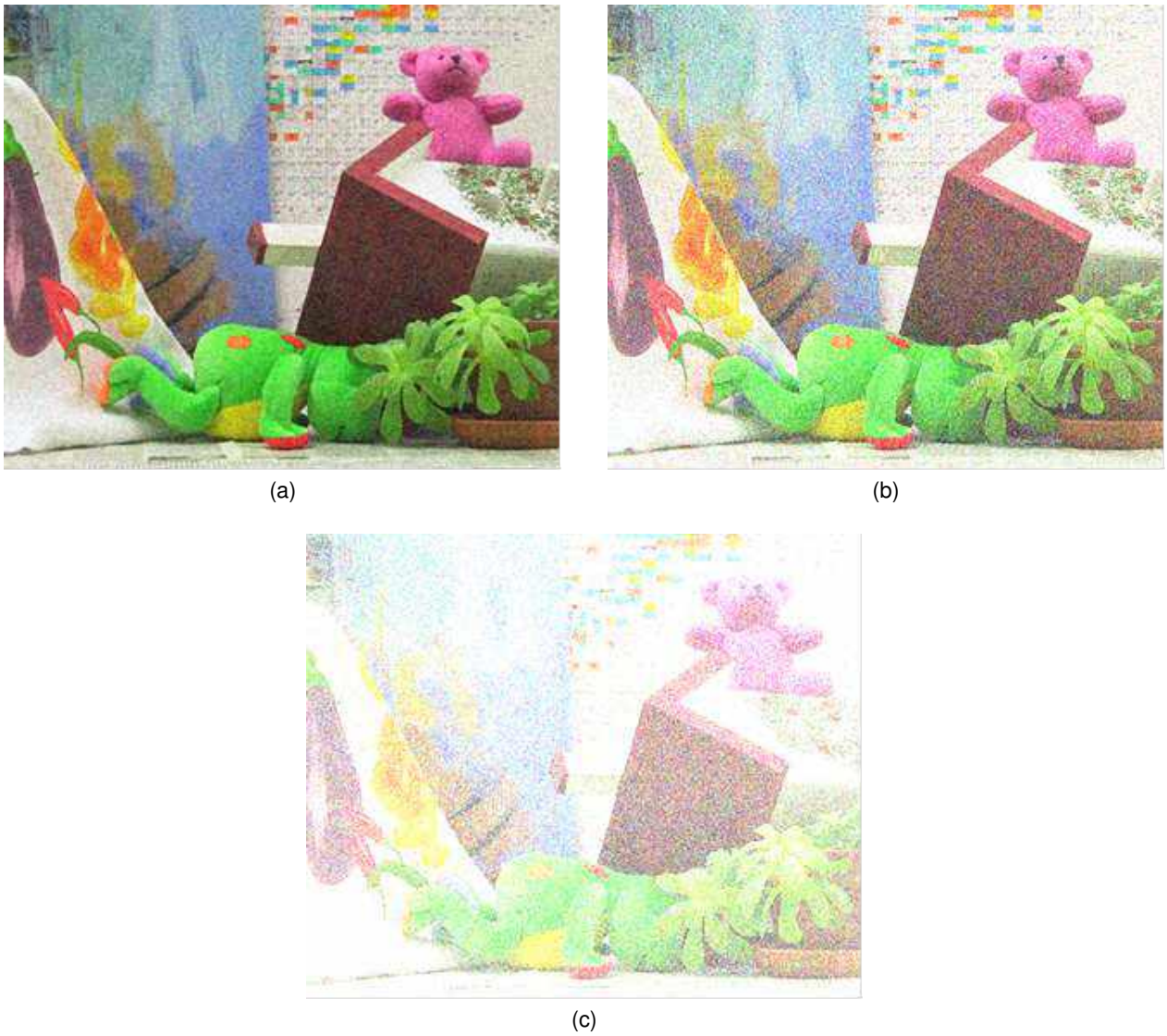
**Fig. 1.** The figure shows the left image (a), the right image (b) and the referenced disparity map (c)

### 2.2 Hierarchical Matching

Hierarchical Matching is one of the most used algorithms in stereo vision [14]. The basic idea of Hierarchical matching is the construction of pyramid of images.

Therefore, we reduce the pair of images: the top of the pyramid corresponds to the lower resolution and the base of the image corresponds to the original of image.

For example: if the original image is 2048×2048 pixels, we reduce the resolution to 16×16. In this case, the pyramid contains eight levels. To move from one level to other, we reduce the image by averaging the pixel values in the square $N \times N$. To compute the disparity map, we start from the top of the pyramid and lower resolution images are matched, we repeat this process until the higher resolution images, hence, obtaining the final disparity map [3, 12].



(a) (b)

(c)

**Fig. 2.** The figure shows the left image (a), (b) and (c) with $0,1$ variance, $0,2$ variance and (c) $0,3$ variance

### 2.3 Correlation Methods

The correlation techniques calculate the resemblance between two images. We consider a correlation window around the matched pixel. The correlation window is compared with a similar window in the right image at all along the epipolar line. Each movement of the window in the right image, a correlation index is calculated. The movement that minimizes the correlation index is retained as disparity. The correlation methods can be classified into two categories: Cross-Correlation and Distances measures. The values obtained by the NCC are in $[0,1]$ and the ZNCC are in $[-1,1]$.

ZNCC is very used in the literature and it corresponds to the classical linear correlation in statistics [5]. The distances measures are also very used. The disparity values are in $[0,\ \mathrm{DisparityRange}]$ [10].

**Table 2.** The results obtained by the correlation methods. (D. P. is the Dynamic Programming and H. M. is the Hierarchical Matching)

| Methods | MAE | MRE | PCM | PWM |
|---|---|---|---|---|
| SAD | 1,79 | 0,015 | 96,69 | 3,31 |
| ZSAD | 0,95 | 0,010 | 98,28 | 1,72 |
| LSAD | 1,03 | 0,011 | 98,11 | 1,89 |
| SSD | 1,58 | 0,013 | 97,06 | 2,94 |
| ZSSD | 1,16 | 0,011 | 98,06 | 1,94 |
| LSSD | 1,26 | 0,012 | 97,85 | 2,15 |
| NCC | 1,24 | 0,012 | 97,87 | 2,13 |
| ZNCC | 0,91 | 0,009 | 98,30 | 1,70 |
| SubPixel | 3,69 | 0,031 | 93,56 | 6,44 |
| D. P. | 2,61 | 0,021 | 95,56 | 4,44 |
| H. M. | 8,67 | 0,047 | 83,82 | 16,16 |

## 3 Evaluation Criteria

In this study, we propose to use four performance measures to evaluate the disparity provided by the algorithms. The first one is the mean absolute error which is the mean of the absolute error between the theoretical disparity map and the computed disparity map. The mean absolute error is defined as follows:

$$\mathrm{MAE} = \frac{1}{N_i \times N_j} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \left| d_c(p_l^{i,j}) - d_t(p_l^{i,j}) \right|, \quad (1)$$

where $d_c(p_l^{i,j})$ is the computed disparity map $d_t(p_l^{i,j})$ and is the theoretical disparity map. Also, we consider the mean relative error that is given by:

$$\mathrm{MEE} = \frac{1}{N_i \times N_j} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \left| \frac{d_c(p_l^{i,j}) - d_t(p_l^{i,j})}{d_t(p_l^{i,j})} \right|. \quad (2)$$

Moreover, we calculate the percentage of correct matches:

$$\mathrm{PCM} = \frac{C}{N_i \times N_j}. \quad (3)$$

And the percentage of wrong matches:

$$\mathrm{PWM} = \frac{W}{N_i \times N_j}, \quad (4)$$

where $C$ is the number of pixel $2$ and $W$ is the number of pixels $> 2$. It is very important to note that the occlusions ($d_c(p_l^{i,j}) = 0$ and $d_t(p_l^{i,j}) = 0$) are not taken into account in the calculation of MAE, MRE, PCM and PWM. The computational time is also used as an evaluation criterion. It allows to show the efficient of each technique in term of timing.

## 4 Experimental Results

In this section, the performance evaluation of each stereo matching algorithm is presented. We conducted the experimentation in terms of:

- Mean absolute error,
- Mean relative error,
- Percentage of correct matches,
- Percentage of wrong matches,
- Computational time.

We consider other important evaluation criteria which is the visual aspect of disparity map. The different disparity estimate algorithms are implemented on a PC with an Intel i32.13 Ghz, 4 GB RAM by using the Matlab and we use the computer vision toolbox for the 3D reconstruction.

The experimentation is conducted on the Middlebury image called “teddy” with a size of $450 \times 375$ pixels and $72$ pixels/inch for the resolution. The ground truth for the images are knows[1]. We use a $9 \times 9$ for the correlation window size and $[0, 50]$ for the disparity range.

We also consider the SAD for the dynamic programming and hierarchical matching. Figure 1 shows the left and the right images with the theoretical disparity map. The aim of this study is to evaluate the performance of matching algorithms and show their efficiency on different noise level.

[1] www.middlebury.edu/stereo/data.html

ISSN 2007-9737



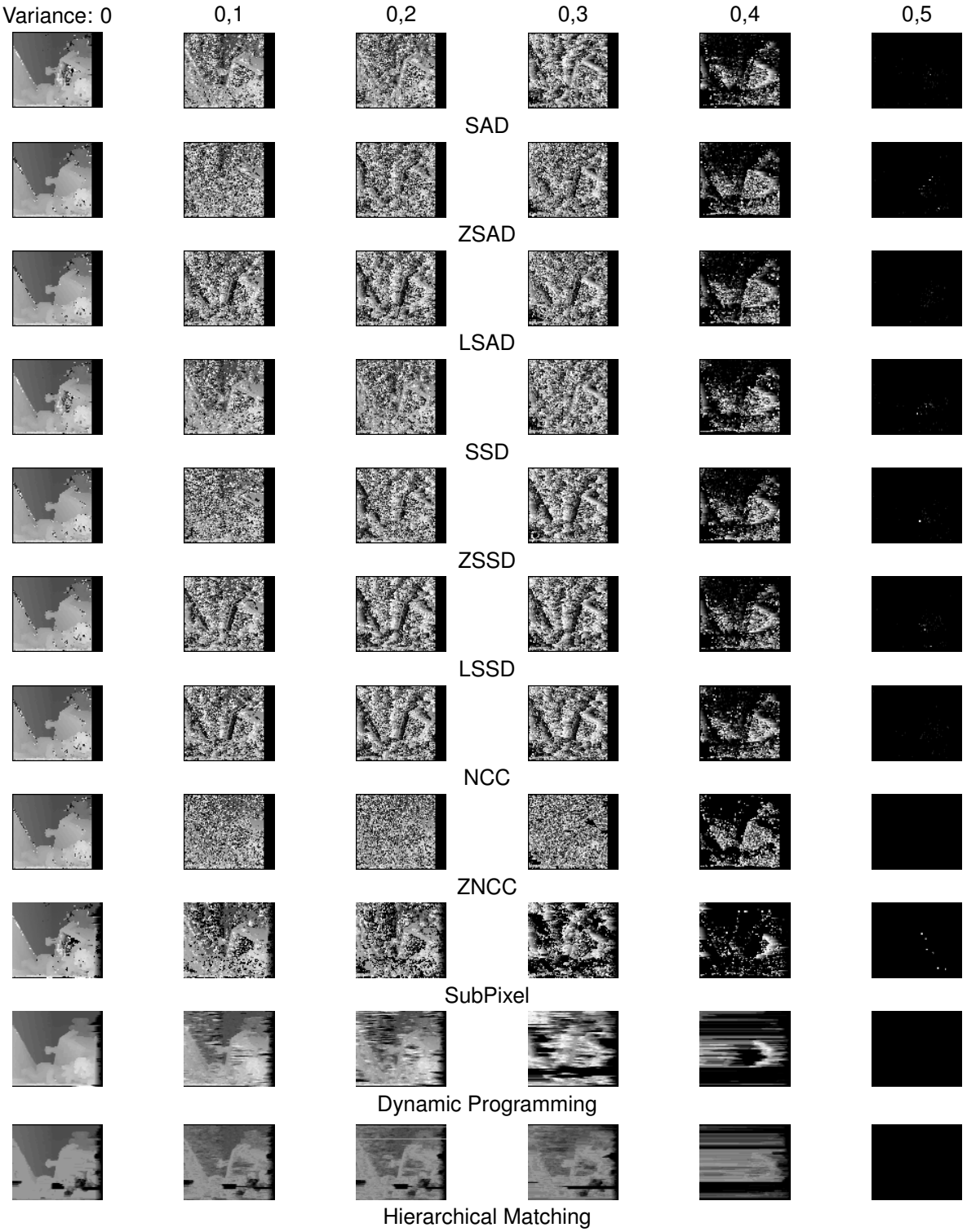



**Fig. 3.** The visual result of stereo matching. Disparity maps computed with different matching algorithms

ISSN 2007-9737



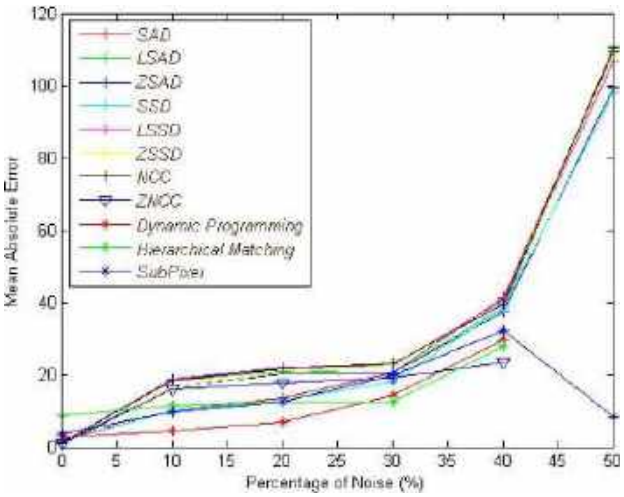



**Fig. 4.** The mean absolute error VS the percentage of noise obtained by each algorithms

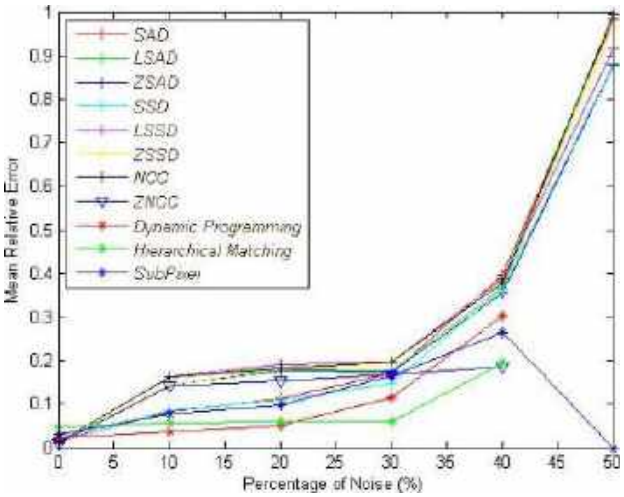



**Fig. 5.** The mean relative error VS the percentage of noise obtained by each algorithms

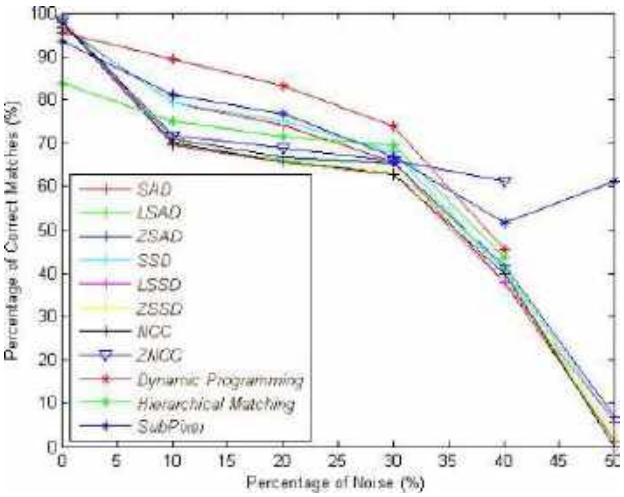



**Fig. 6.** The percentage of correct matches VS the percentage of noise obtained by each algorithms

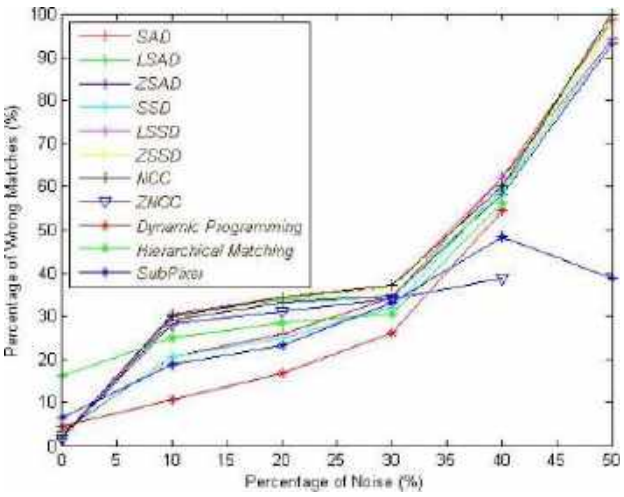



**Fig. 7.** The percentage of wrong matches VS the percentage of noise obtained by each algorithms

We use the gaussian noise with different variance : $0,1 - 0,2 - 0,3 - 0,4$ and $0,5$. Figure 2 illustrates the different left images obtained by using different noise levels. Note that the right images have the same noise levels. In the figure 3, we show the visual results of stereo matching obtained by the matching algorithms defined above. The analyse of the visual results obtained by the different stereo matching algorithms shows that the numbre of occlusions points increase when the variance of the noise change. Therefore, if the left and the right images contain a high noise level, the number of occulions point increase significatly; also; with $0,4$ and $0,5$ variances of gaussian noise, we observe clearly that the disparity map is black. This means that the scene of left image is occulted. Visually, the dynamic programming and the hirearchical matching algorithms provide a smooth disparity map. The smoothing of the disparity map give a good depth map with noise

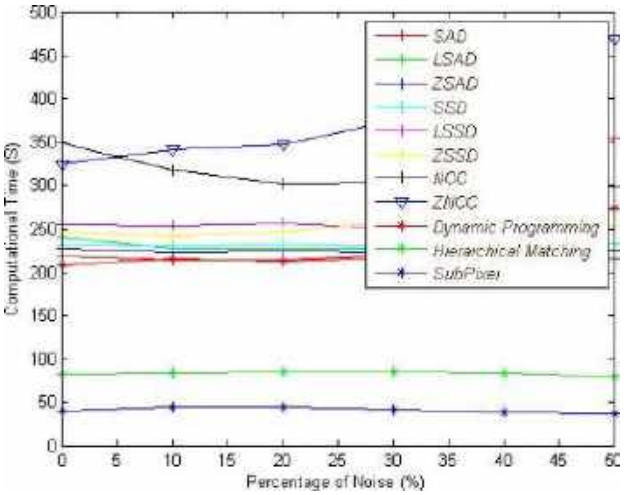



**Fig. 8.** The computational time VS the percentage of noise obtained by each algorithms

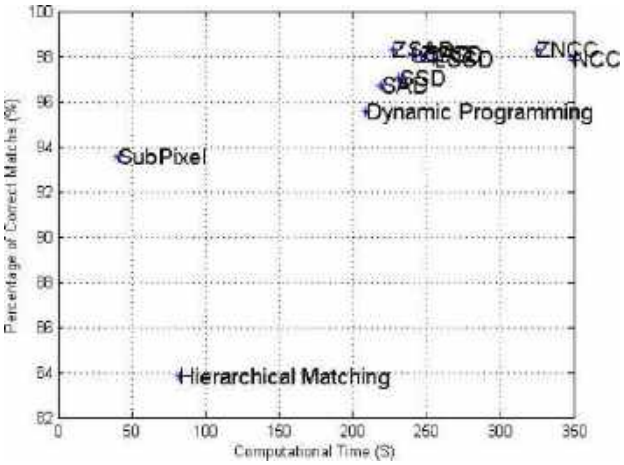
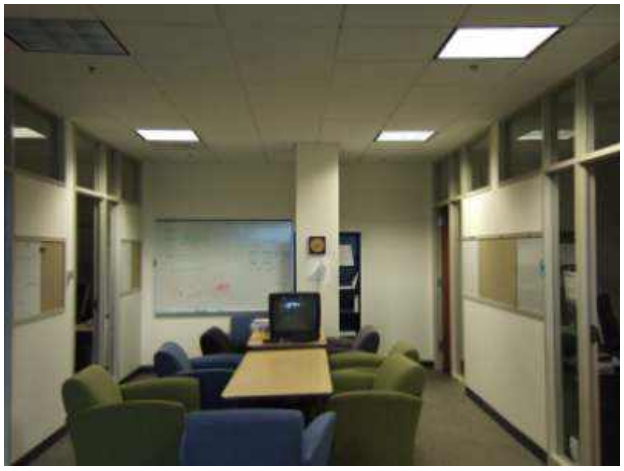



**Fig. 9.** The computational time VS the percentage of correct matches



Left image

Right image

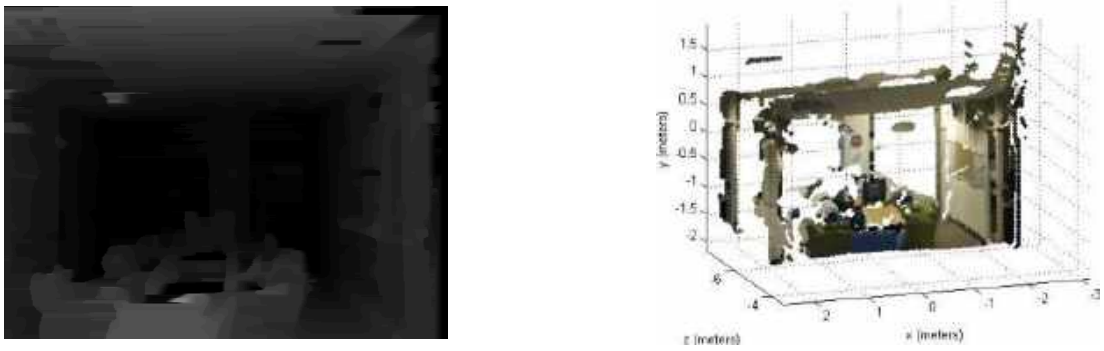
**Fig. 10.** The left image and right image

images. ZNCC technique gives a low quality of disparity map when the variance of noise is $0,3$ and $0,4$ compared to others. To validate these results, numerical results are considered. Table 2 describes the numerical result obtained by the above algorithms. We calculate the MAE, MRE, PCM and PWM by using the left and the right images without noise. In term of mean absolute error, the low value is recorded for the ZNCC technique with $0,009$ for the mean relative error and $98,30\%$ of correct matches.

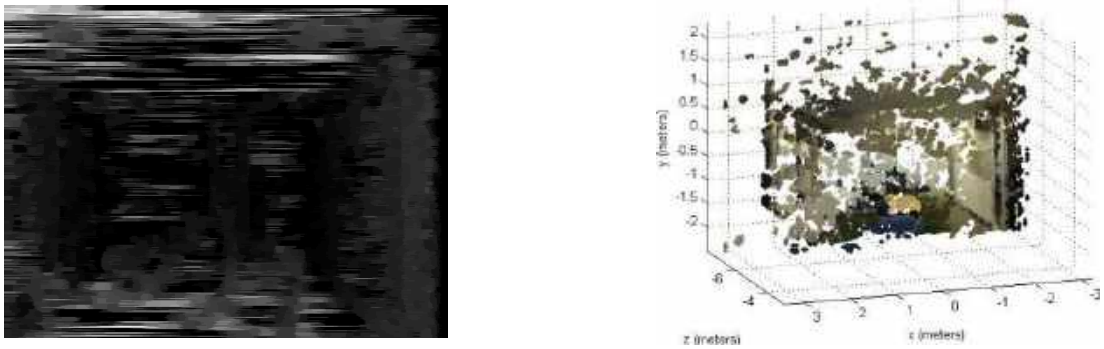
We remark that ZNCC technique provide a good results with images without noise but, it is not very perfect with noise images. The high percentage of correct matches is also recorded for the ZSAD, ZSSD with an advantage of ZSAD $98,28\%$ of correct matches. The hierarchical matching algorithm provides a bad results, we estimate that this results is due to the fact that the low resolution of the image has given a bad disparity map. Figures 4, 5, 6 and 7 illustrate the MAE, MRE, PCM and PWM vs the percentage of

ISSN 2007-9737

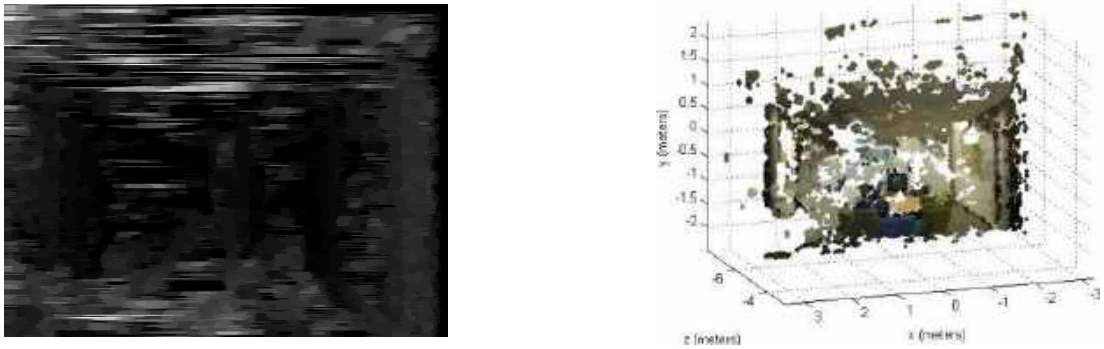




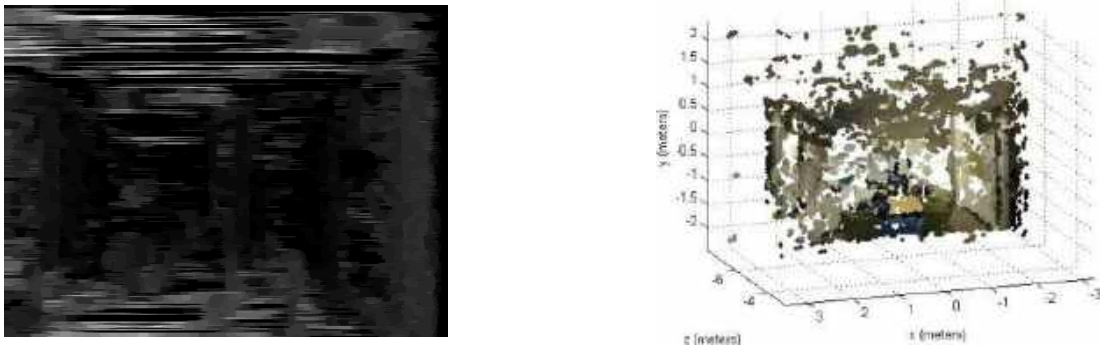
Disparity map and 3D reconstruction without noise images



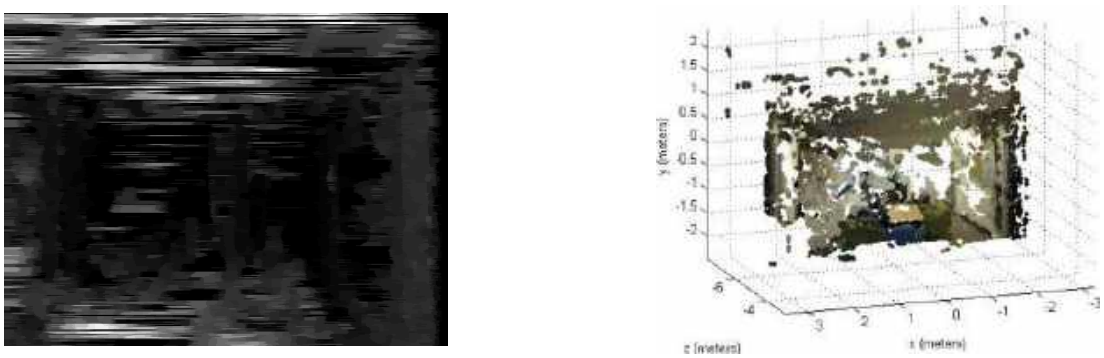
Disparity map and 3D reconstruction with variance 0,1 of noise



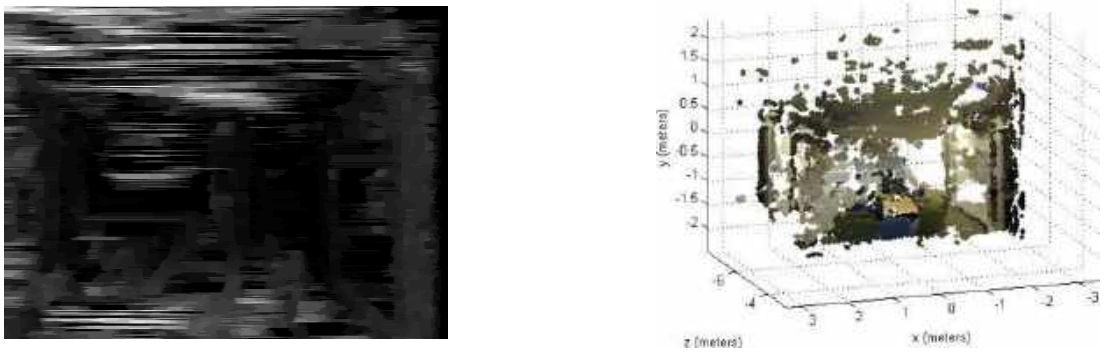
Disparity map and 3D reconstruction with variance 0,2 of noise



Disparity map and 3D reconstruction with variance 0,3 of noise



Disparity map and 3D reconstruction with variance 0,4 of noise



Disparity map and 3D reconstruction with variance 0,5 of noise

**Fig. 11.** Disparity map and 3D reconstruction from the dynamic programming algorithm at different noise levels

gaussian noise applied to the left and right images. Figure 4 describes the different percentage of noise used in this study. It clearly visible that the lower value is recorded for dynamic programming in each noise level.

There is no value for hierarchical matching, ZNCC and pynamic programming for $50\%$ percentage of noise, this means that the disparity map when the left and the right images is noised $(50\%)$ are black (occulted scene).

The same remark is observed in figure 6 and 7. We cleary show that the percentage of correct matches is higher in each noise levels when we use the dynamic programming algorthim.

Generally speaking, the dymanic programming algorithm provide a very good results with noise images. It is very interesting to evaluation the efficiency of these matching algorithms in term of computational time.

Figure 8 show the computational time of each methods versus the percentage of noise. The lower computational time is recorded for the subpixel and hierarchical matching. The slow method is ZNCC and NCC. To show the efficiency of the algorithms, we plot the percentage of correct matches versus the computational time.

The most efficient algorithm is the one with small computational time and with a high percentage of correct matches. In other words, the algorithm which appear further to the top left of the figure is the better one.

Finally, it is noted that the ZNCC gives better results when used with no noisy image, whereas, the dynamic programming provides better results when used a noisy image.

This is why, we choose dynamic programming algorithm for $3$D reconstruction. We use the computer vision matlab toolbox to realize the $3$D reconstruction.

We consider the "hallway" images of computer vision matlab toolbox with know intrinsics matrix:

| | | |
|---|---|---|
| Focal_length_x | = | 409,4433 |
| Focal_length_y | = | 416,0865 |
| Camera_center_x | = | 204,1225 |
| Camera_center_y | = | 146,4133 |

Following figures represent the left image and the right image used in the 3D reconstruction[2].

## 5 Conclusion

In this paper, we analyzed and evaluated the performance of some stereo matching algorithms. We proposed a comparison protocol for the efficient and comparing their efficiency and performances in term of: absolute and relative errors, accuracy of matches by calculating the percentage of correct and wrong matches.

Also, we consider the computational time and the robustness of each algorithm to noise. The experimentation has been conducted by using the teddy images obtained from middle bury data set. We used different percentage of Gaussian noise applied to the left and right images and we calculate the disparity maps. Finally, we proposed a $3$D reconstruction using the better disparity map obtained from the methods.

The results has shown that the ZNCC has given a very satisfactory results when it was used with images without noise. Dynamic programming has provided a good results, it is a very robust against noisy images. In term of computational time, the low time was recorded for subpixel estimation and hierarchical methods.

## References

1. **Ali, H., Nema, B. (2009).** Multi purpose code generation using fingerprint images. The International Arab Journal of Information Technology, Vol. 6, pp. 141–145.
2. **Kolesnik, M. I. (1993).** Fast algorithm for the stereo pair matching with parallel computation. Computer Analysis of Images and Patterns, pp. 533–537. DOI: 10.1007/3-540-57233-3_70.
3. **Koschan, A., Rodehorst, V., Spiller, K. (1996).** Color stereo vision using hierarchical block matching and active color illumination. Proceedings of 13th International Conference on Pattern Recognition, Vol. 1, pp. 835–839. DOI: 10.1109/ICPR.1996.546141.
4. **Leung, C., Appleton, B., Sun, C. (2008).** Iterated dynamic programming and quadtree subregioning for fast stereo matching. Image and Vision Computing, Vol. 26, No. 10, pp. 1371–1383. DOI: 10.1016/j.imavis.2007.11.013.
5. **Lhuillier, M., Guo-Quan, L. (2004).** Reconstruction quasi-dense de modeles 3d partir d'une sequence d'images. Actes du Congres AFRIF-AFIA Reconnaissance des Formes et Intelligence Articielle, RFIA, Vol. 2, pp. 895–904.
6. **Ouali, M. (2012).** A Markov random field model and method to image matching. International Arab Journal of Information Technology, Vol. 9, No. 6, pp. 520–528.
7. **Ouali, M. (2012).** Performance evaluation of stereo matching algorithms in the lack of visual features. International Journal of Computer Applications, Vol. 53, No. 5, pp. 7–11. DOI: 10.5120/8415-0636.
8. **Park, C. S., Park, H. W. (2001).** A robust stereo disparity estimation using adaptive window search and dynamic programming search. Pattern Recognition, Vol. 34, No. 12, pp. 2573–2576. DOI: 10.1016/s0031-3203(01)00016-4.
9. **Pissaloux, E. E., Le-Coat, F., Bonnin, P. J., Bezencenet, G., Durbin, F., Tissot, A. (1997).** Very fast dynamic programming-based parallel algorithm for aerial image matching. Automatic Target Recognition VII. DOI: 10.1117/12.277122.

[2] www.mathworks.com/products/computer-vision/

10. **Seong-Ku, J., Lee-Mu, K., Uk-Lee, S. (1998).** Multi-image matching for a general motion stereo camera model. Proceedings 1998 International Conference on Image Processing, Vol. 2, pp. 608–612. DOI: 10.1109/icip.1998.723543.
11. **Szeliski, R. (2022).** Computer vision: Algorithms and applications. Springer International Publishing. DOI: 10.1007/978-3-030-34372-9.
12. **Thevenaz, P., Ruttimann, U. E., Unser, M. (1998).** A pyramid approach to subpixel registration based on intensity. IEEE Transactions on Image Processing, Vol. 7, No. 1, pp. 27–41. DOI: 10.1109/83.650848.
13. **Veksler, O. (2005).** Stereo correspondence by dynamic programming on a tree. Vol. 2, pp. 384–390. DOI: 10.1109/CVPR.2005.334.
14. **Won, K. H., Jung, S. K. (2011).** hSGM: Hierarchical pyramid based stereo matching algorithm. pp. 693–701. DOI: 10.1007/978-3-642-23687-7_62.

*Article received on 19/12/2023; accepted on 16/04/2024.*
** Corresponding author is Seyyid Ahmed-Medjahed.*

ISSN 2007-9737

# Neural-Combinatorial Classifiers for Arabic Decomposable Word Recognition

Zeineb Zouaoui*, Imen Ben-Cheikh, Mohamed Jemni

University of Tunis,
National High School of Engineering of Tunis,
Department of Computer Science,
Tunisia

{zeineb.zouaoui, imen.becheikh}@gmail.com, mohamed.jemni@alecso.org.tn

**Abstract.** Recognition tools and techniques for Arabic script are still under development due to the topological ambiguities and inflectional nature of this language. In this regard, this paper presents an approach based on a combinatorial optimization technique incorporating convolutional neural networks for Arabic word recognition. We handle a wide vocabulary of Arabic decomposable words. We adopt a design that resembles a molecular cloud with words structured according to their roots and patterns. This conception fits well with the Arabic linguistic philosophy of building words from their roots. Hence, each sub-vocabulary represents a sub-cloud, encompassing neighboring words derived from the same root and following different patterns and forms of derivation, inflection and agglutination (proclitic and enclitic). Hence, each sub-vocabulary represents a sub-cloud, encompassing neighboring words derived from the same root and following different schemes and forms of derivation, inflection and agglutination (proclitic and enclitic). Accordingly, as a first step, we have used a recognition approach based on the metaheuristic method of simulated annealing (SA). In a second work, we implemented the SA algorithm by integrating linguistic knowledge. Extending this work, we choose to integrate a convolutional neural network into the recognition process of the SA algorithm to benefit from the advantages of both methods. To conduct our experiments, which yielded promising results, we use a corpus of Arabic words including samples and agglutinated words from the APTI database.

**Keywords.** Convolutional neuronal network, combinatorial optimization, simulated annealing, morphological characteristics, Levenshtein distance.

## 1 Introduction

Text recognition remains a very active area of research. Which aims to read texts efficiently, similarly to human reading process. The recognition process is challenging mainly due to the adopted approach (structural, analytic), script language and vocabulary size. These ongoing challenges handling with Arabic language are due to its cursive nature, script types and morphological peculiarities.

Vocabulary size (reduced or large) also influences the performance of a recognition system as well as the adopted method (stochastic, neural, etc). Although there are several approach classes for characterizing and recognizing Arabic scripts, the obtained results did not yield performances similar to those achieved for other scripts, like Latin.

To face these challenges, a robust methodology becomes necessary. Hence, they are two radically different approaches, machine learning and combinatorial optimization can be used together to serve a common goal and solve the same problem [1].

On one hand, training algorithms are able to learn and generalize on the basis of unstructured or unformalized information. On the other hand, combinatorial optimization methods are frequently proposed to solve pragmatically artificial intelligence problems with large solution spaces. We attempt then to hybridize these two types of approaches in order to develop new methods of

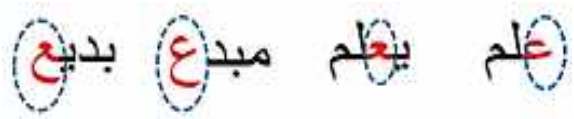


**Fig. 1.** Samples of different shapes of the same letter (ع) **/E/** in different positions

resolution and novel perspectives by taking into account the major challenges associated with the complexity and specificity of Arabic scripts.

The primary contributions of this paper can be summarized as follows: (1) To focus on incorporating linguistic knowledge into the recognition process, thanks to the importance of lexical and syntactic information in written words. (2) To benefit from combinatorial optimization and machine learning methods in order to propose an approach that handle with large vocabulary of Arabic decomposable words (derived from roots) presenting various morphological features (derivational, inflectional and agglutinative).

The remainder of this paper is structured as follows: Section 2 presents a short morphological and topological analysis of Arabic script. Section 3 describes natural language processing for Arabic script recognition.

Section 4 displays a brief description of combinatorial optimization techniques. A convolutional neural network is illustrated in section 5. Section 6 reviews the related works suggested in the literature. Section 7 introduces our proposed approach. Then, section 8 reports our experiments by focusing on the major findings. Finally, section 9 summarizes the paper and highlights major directions for future research.

## 2 Arabic Script Analysis

In this section, we analyses Arabic words by taking into account two major aspects, namely the linguistic aspect and the morphological aspect.

### 2.1 Topological Peculiarities

- The Arabic language has a huge vocabulary, letters are mostly connected, and most words consist of more than one sub-word, called PAW (Pieces of Arabic Word).
- An Arabic script is characterized by the presence of a horizontal baseline or a reference line. This latter is the place of horizontal ligature characters of the same string.
- Arabic characters are written cursively from right to left in both manuscript and print.
- The Arabic alphabet comprises 28 basic characters, 16 of these characters include one, two, or three diacritical dots in their various forms.
- These dots distinguish between characters of the same body. They can also be located above or below the body of the basic character.
- An Arabic letter has different forms according to their positions in the word: at the beginning, in the middle or at the end of the word (see Fig.1).

These particularities distinguish Arabic from other languages. Moreover, their absence can cause some problems. For instance, the absence of diacritical dots and vowels causes confusion between letters of the same shape. Fig.2 illustrates samples of these problems.

The word "**جمل**" **/jamal/**[1] with diacritics that means a camel -differs from the word "**حمل**" **/Hamal/** without diacritics- that means a lamb. Moreover, an Arabic text without vowels is highly ambiguous and the correct functions of words cannot be distinguished.

For example, the word "**علم**" **/Elm/** (without voyellation) is ambiguous as it has different meanings depending on its voyellation. It can mean science "**عِلْمٌ**" **/EilomN/**, flag "**عَلَمٌ**" **/EalamN/**, the past tense verb (he knew) "**عَلِمَ**" **/Ealima/** or "**عُلِمَ**" **/Eulima/** (has been known).

Taking another example, the absence of capital letters in the Arabic word "**جميلة**" **/jamiylap/** can generate ambiguity in determining its function: whether an adjective, a proper noun? It can mean either the adjective "beautiful" or the proper noun "Jamila".

[1] Transliteration is coded following Buckwalter transliteration: https://www.ipabwat.com/

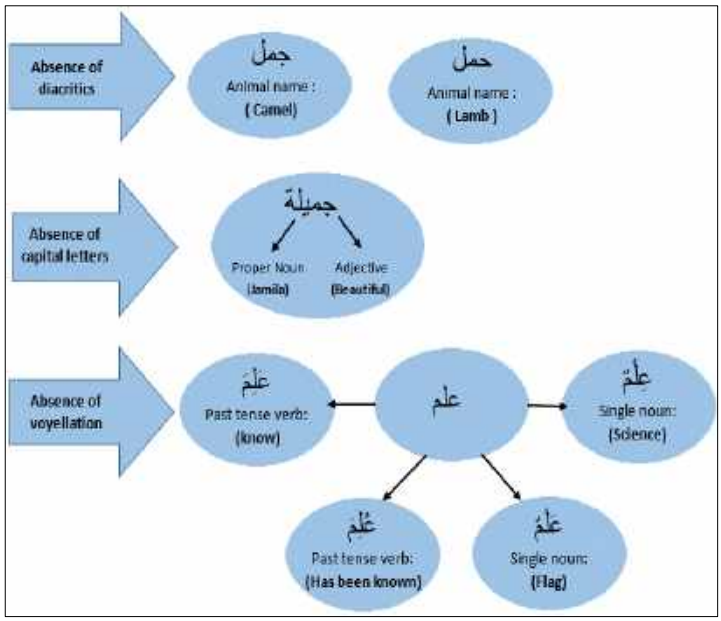



**Fig. 2**. Examples of ambiguities related to the absence of diacritics, voyellation and capital letters in arabic words

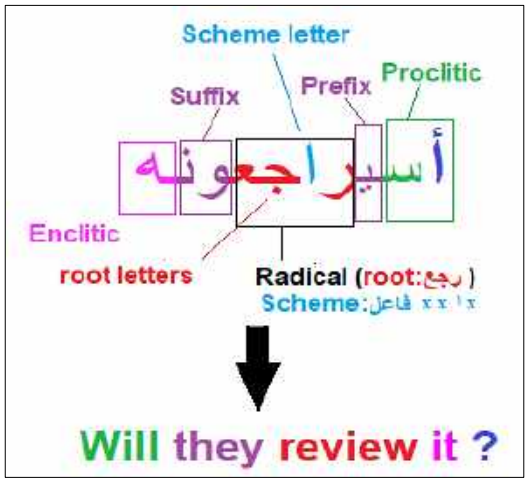



**Fig. 3.** The agglutinative derivation of the single word "**أسيراجعونه**" **/>asayuraAjiEuwnahu/**

## 2.2 Morphological Peculiarities

Morphological analysis is a challenging task in Arabic natural language processing compared to other languages. This is due to the diversity and complexity of words linguistic characteristics.

For instance, (see Fig.3), a full English sentence can be translated into a single Arabic word; such as the sentence **"Will they review it?"** that can be replaced by the word "**أسيراجعونه**" **/>asayuraAjiEuwnahu/**. Furthermore, an Arabic word is either decomposable or non-decomposable. Non-decomposable words cannot be divided into affixes (prefixes, infixes and suffixes) and roots. This word category includes pronouns, proper nouns, numbers, country names, and particles (equivalent to adverbs, prepositions, conjunctions, etc.).

Various non-decomposable words are made up of new non-Arabic words that have found their way into the language by taking into account some terms associated with modern technology [2].

A decomposable word, however, results from its root derivation according to a conjugated scheme. A root is purely consonantal; that is to say it is formed by adding a sequence of three, four, or even five consonants (for nouns) in order to build the base of the word. A root is a salient element in derivative languages. Indeed, each root corresponds to a semantic field.

Thanks to different schemes, we can generate a family of words in each semantic field. The scheme is composed of three different consonants ف /f/, ع **/E/**, and ل **/l/**, which are vocalized and can be augmented by adding other letters, namely prefix, suffix, and infix. The root derivation gives a word its radical meaning according to the short scheme as explained in Fig.3.

This paradigm allows the generation of new words by adding prefixes that precede a root base, by injecting infixes that appear between the basic letters and by adding suffixes coming at the end. This morphological diversity is also clear in the inflectional process. In fact, a radical undergoes inflected conjugations that are able to change the prefixes and suffixes and/or adjoin additional letters. In this way, the elements of inflectional conjugations of Arabic comprise the personal pronoun, time, gender, number, function (accusative or nominative), etc.

Arabic language morphology is also characterized by its agglutinative aspect. Agglutination can be divided into two main kinds: proclitics and enclitics [3]. A proclitic can be either simple morphemes of one letter (coordinators, conjunctions, and prepositions) or complex morphemes of multiple letters (the combination of simple proclitics).

The enclitic is a supplement pronoun that can be single or double and is linked to the word that precedes it. Fig.3. describes the derivation process of the agglutinated word **سيراجعونه /sayuraAjiEuwnahu/**. Which results from the conjugation of the root **رجع /rajaEa/** according to the scheme **فاعل /faAEala/** (x x ا x; where x refers to one of the three consonants) with the pronoun **هم /hum/** (they: plural masculine) by adding both

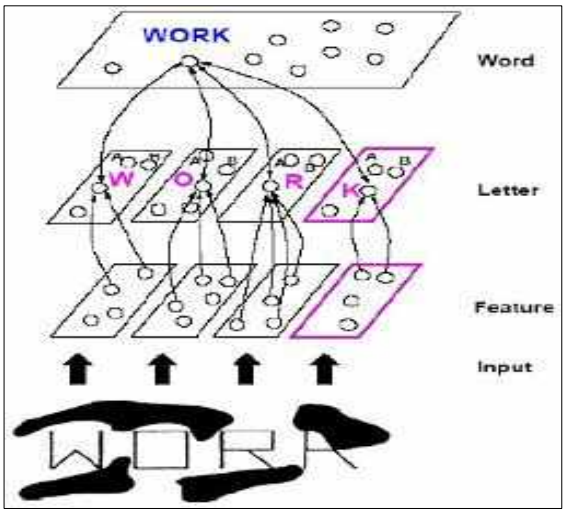



**Fig. 4.** A perceptual model inspired by human vision system [8]

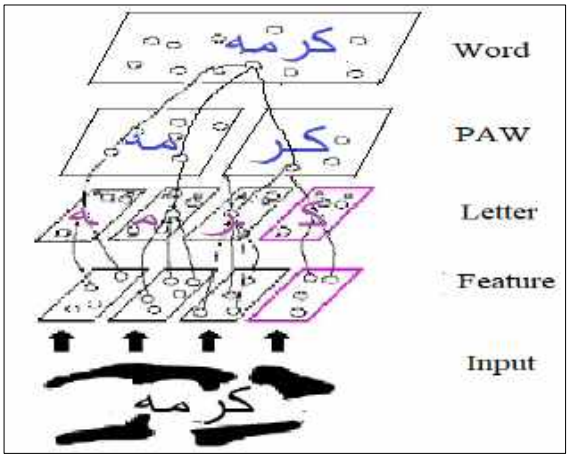



**Fig. 5.** A perceptual model influenced by the word superiority effect

the proclitic س **/s/** (designating the future tense) and the enclitic ه **/h/** (designating the object complement: him).

## 3 Natural Language Processing

Pattern recognition is the automation of artificial perception tasks performed by the human sensory system and brain. It aims to classify entities into categories on the basis of observations made on them [4].

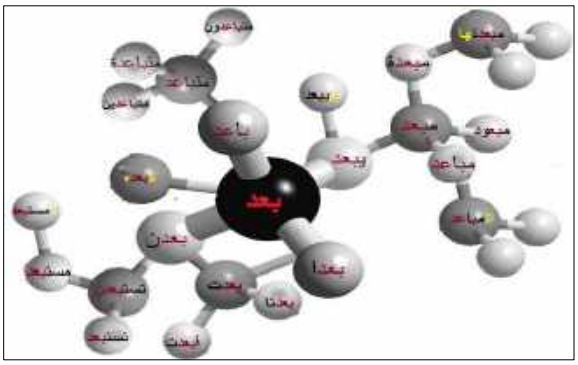


**Fig. 6** Parts of the sub-cloud corresponding to the root "**بعد**" **/baEada/** (i.e. to go far)

**Fig. 7** CNN-Molecular cloud: the association of several sub-clouds (black balls symbolize sub-cloud cores corresponding to the roots

In this respect, psychoanalytic studies show that humans memorize letters standing for the whole word instead of sequences of separate letters. Therefore, letters of words can be memorized better than letters in meaningless sequences. The "Word Superiority Effect" concept, proposed by Mc Clelland and Rumelhart [5, 6], was inspired by this human reading perception (see Fig.4). It permits a layered representation of the word, from the local to the global layer and viceversa.

According to [6], this model is applicable to Arabic script recognition on condition that an intermediary global level, known as the level of the pseudo word (PAW), is added as shown in Fig.5. Therefore, word recognition can be guided by the meaning of different possibilities of PAW combinations.

The importance of the "Pseudo-Word Superiority Effect" decline not only from the primary function of PAWs in the Arabic language, but also from the analogy with the "Word Superiority Effect", since PAWs lead to not totally cursive writing.

Indeed, some Arabic letters change their shapes to maintain cursive continuity of the writing, where others maintain their shape including space between letters and introducing, thus, non-cursiveness into writing. In the same direction, the integration of linguistic information in the recognition process is one of the most promising research approaches.

In this context, studies in references [3,7], which focused on the "Word Derivation Effect" by integrating linguistic information such as roots and patterns in the Arabic word recognition process, yielded some interesting results. Accordingly, we generally estimate that word recognition can occur at several independent, but complementary, levels, such as root recognition, pattern recognition, agglutination recognition, recognition conjugate elements, etc.

Then, we assume that the human reading process is also characterized by the notion of "Word Morphology Effect". To substantiate this, we will prove that the incorporation of morphological knowledge in a perceptual model helps us to recognize a large vocabulary of Arabic words. To this end, the suggested approach combines Arabic morphology with the human reading process.

## 4 Combinatorial Optimization

Combinatorial optimization (CO) problems consist in choosing the best solution out of a very large, but finite, set of solutions. Hence, solving a CO problem means finding an optimal function from a very large but finite number of choices. To solve CO problems, it is necessary to choose the most appropriate method, (either exact or heuristic method).

Exact methods explore the whole search space in order to find at least one of the optimal solutions for a given problem. These methods have mostly been used on small problem instances. On the contrary, heuristic methods explore a subset of the search space in order to find a practical solution

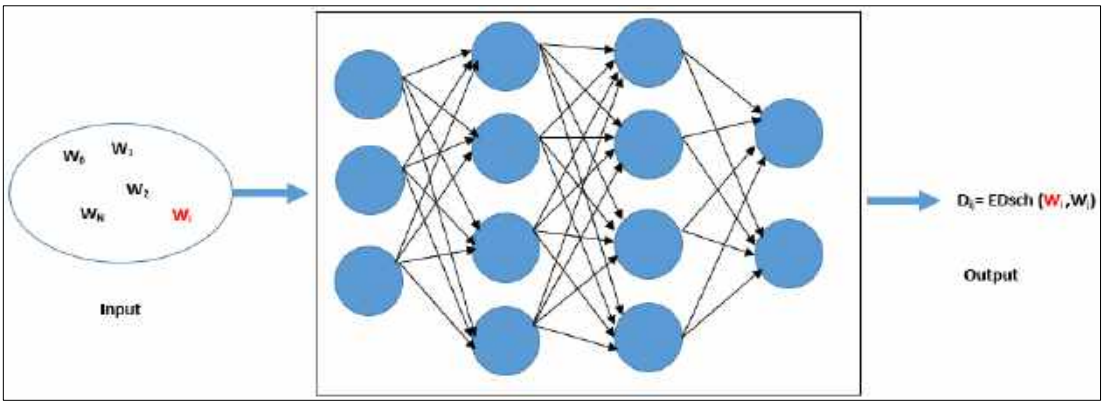



**Fig. 8** CNN architecture

within a reasonable time. They can be often applied to all types of problems. Moreover, heuristic methods are divided into two sub-classes, namely meta-heuristic approaches and constructive approaches.

The former search for the global optimum over a set of local optimums, while the latter, also known as iterative methods, build a complete solution step by step. Metaheuristics fall into two major categories: evolutionary algorithms (e.g., genetic algorithms, etc.) and neighborhood methods (e.g., Tabu Search, Simulated Annealing, Hill Climbing, etc). The former deal with sets of solutions at the same time to quickly find the optimal solution.

The latter, however, start their search from a simple initial solution that will be modified and improved through research in order to achieve an optimal solution.

Finally, the choice of an appropriate combinatorial method depends on some characteristics, such as the problem complexity, cost evaluation, the existence of constraints, the desired results (the number of solutions and their properties), etc.

This information allows us to select the best technique to be used from the numerous available alternatives. Despite their differences, combinatorial optimization and machine learning methods aim to reach a common goal and solve the same problem. The hybridization of combinatorial optimization and learning methods leads to new methods of resolution and novel perspectives [1]. In this regard, hybridization consists in fully exploiting each method's strengths in order to obtain the best performing solution.

## 5 Convolutional Neural Network

A convolutional neural network (CNN), originally proposed by LeCun [9], is a neural network model. It combines three key architectural ideas: local receptive fields, shared weights, and spatial subsampling. CNN architecture is composed of three main layers:

- Convolutional layer applies a filter to an input image. The convolution value is calculated by taking the scalar product of the corresponding kernel values and the channel matrices.
- Pooling layer is generally placed after the convolutional layer. Its main utility consists in reducing the spatial dimensions (width/height) of the input volume for the next convolutional layer. In the pooling layer, we use a sliding window moving over the input in large steps. With each movement of the window, representative values of data involved in the window are detected.
- Max pooling takes the maximum value in each window. It is preferred over other techniques thanks to its performance characteristics.
- Fully connected layers are fully connected to the previous output layer. They are commonly used at the end of a CNN to connect each hidden layer to the output layer and then build the required number of outputs.

```
W = select a random word from the cloud (current solution)
EDsch (W) = New Edit distance between the unknown Word and the current solution W
Set the initial temperature initial_T
While (temperature >0 and non-convergence)
        Neighbor = select the best neighborhood word solution
        Calculate delta = EDsch (neighbor) – EDsch (W)
        If (delta <= 0)
            W = neighbor
        Else
            Select new neighbor with probability e^-(delta/t)
        End If
        Decrease the temperature
End While
Output the final solution
```

**Fig. 9.** A simulated annealing pseudo algorithm with a new Edit distance [20]

**Table 1.** Correspondence between a physical system and an optimization problem

| Optimization problem | Physical problem |
|---|---|
| Solution | System state |
| Objective function | Free energy (E) |
| Problem parameters | Coordinates of particles |
| Find a good configuration | Find low energy states |
| Global optimum | Ordered stable state |
| Local optimum | Metastable state |
| Local search | Rapid quenching |
| The T parameter | Temperature |

**Table 2.** Examples of classic vs new edit distance calculation on words following different schemes (sch)

| Input | | Classic distance ED | New distance EDsch |
|---|---|---|---|
| يبعد (sch= فعل) | يبتعد (sch=افتعل ) | 1 | 2.9 |
| يبعد (sch= فعل) | مبعد (sch= مفعل) | 1 | 2.6 |
| يبعد (sch= فعل) | يبعده (sch= فعل) | 1 | 1.3 |

The convolutional neural network has various assets. First, feature extraction and classification are integrated into a single structure that is fully scalable.

Second, the network extracts 2D images at multiple dyadic scales. Third, it is relatively invariant to local geometric distortions in the image.

CNN has been used in several applications, such as digit number recognition, face detection and facial recognition.

## 6 Related Works (Linguistic and/or CNN and/or Combinatorial)

Many experiments have shown that human readers, unlike automatic recognitions software, are skilled in combining syntactic, semantic, and morphological analyses. In reading Arabic written expressions, humans rely on the very regular word-form structure. Following this word perception, [10] set forth an affixal approach for Arabic script recognition.

It consists in segmenting words into letters and recognizing their morphological entities. It is an analytical approach to segmentation, which allow authors to integrate linguistic knowledge into the coherence verification of prefixes and affixes, rather than in the recognition process.

In addition, an “affixal approach” was proposed in [2, 7, 3]. Unlike [10], which first recognize letters locally and then words, Markovian and neural classifiers in [2, 7, 3] were suggested in order to first recognize linguistic word concepts globally (prefixes, roots, schemes, etc.).

For instance, [11] put forward two systems for Arabic offline handwriting recognition. The first system uses the connectionist temporal classification (CTC) combined with Bidirectional Long Short-Term Memory (BLSTM) architecture.

The second system, however, is based on cascade CNN and MDLSTM layers. The experiments were carried out on the KHATT database and proved the performance of the two proposed systems.

Reference [12] offered a spotting system for printed and handwritten text line images. It used a hybrid architecture composed of a deep bidirectional neural network, long-term memory and a hidden Markov model (HMM). It exploits not only the robust learning capability of deep neural network representations, but also HMMs sequential modeling ability.

Their system was tested on two Arabic databases (KHATT, PKHATT) and the RIMES Latin database. The experimental results on script identification and keyword extraction confirm the performance of the suggested approach. In a study conducted by [13], a new model was proposed based on deep neural networks for offline Arabic handwritten text recognition (digits, characters and single words).

In fact, the author proposed a supervised convolutional neural network (CNN) model that contextually extracts optimal features and employs batch normalization and dropout regularization parameters.

The model was tested and proven on the MNIST English Isolated Figures database with an accuracy of 99.94% and a precision of 99.68%.

Some researchers have recently attempted to benefit from the advantages of automatic learning in order to solve combinatorial optimization problems. In this context, we can cite the research work of [1], which utilized learning and combinatorial optimization techniques for graph matching, pattern recognition and classification, and computer vision problems.

Moreover, [14] recommended an approach to solve combinatorial optimization problems using neural networks (LSTM) and reinforcement learning. They focused on the traveling salesman problem and trained a recurrent neural network that, given a set of city coordinates, predicts a distribution over different cities permutation.

This approach achieves optimal results on 2D Euclidean graphs with up to 100 nodes.

## 7 Proposal of the Approach

The major objective of this work is to propose an approach for recognizing a large vocabulary of Arabic words, and exploiting the benefits of neural methods and combinatorial optimization techniques. These mechanisms consist of three different steps: In the first step, the vocabulary

**Fig. 10**. Samples of APTI database

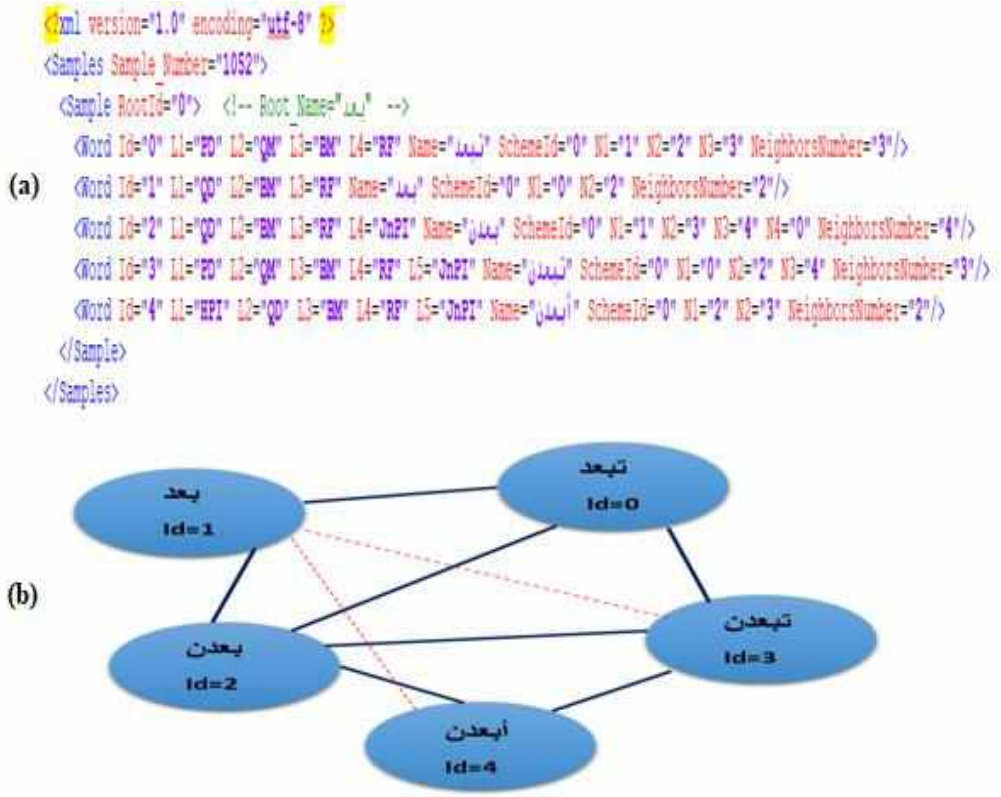



**Fig. 11** Sample of neighbor's corpus: (a) Snippet of word corpus (b) Snippet of word neighborhood

**Table 3.** Some word CNN samples

| | Arabic Word | Transliteration | Score |
|---|---|---|---|
| **CNN_0/0/6/10/4/0** | تبعدن | taboEudona | 99,48% |
| **CNN_0/0/5/11/5/0** | بعدن | baEadona | 99,23% |
| **CNN_0/2/5/10/0/0** | باعدتما | baAEadotumaA | 88,52% |

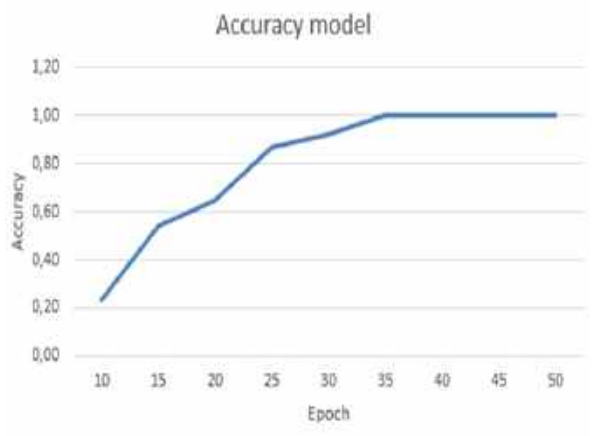



**Fig. 12.** Evolution of training data

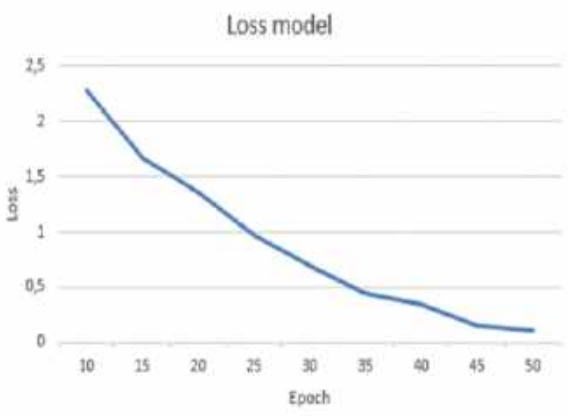



**Fig. 13**. Loss function on training data


**Fig. 14**. Evolution of recognition rates depending on the cooling speed

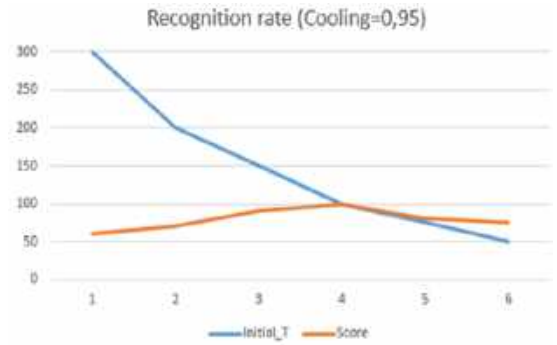



**Fig. 15.** Evolution of recognition rates depending on the temperature

structure is performed. Next, the learning step is based on CNN techniques and finally, a recognition step uses the SA combinatorial optimization algorithm. Our approach is, then, hybridizing: NLP + CNN + SA.

### 7.1 Arabic Vocabulary Structuring

In this section, we provide a detailed description of the structure we have chosen for the vocabulary. As first step, our vocabulary of decomposable words is structured as a molecular cloud, whose organization emulates the Arabic linguistic philosophy of word construction around e roots. Each sub-cloud collects neighboring words taken from the same root based on various derivational forms, flexions and agglutinations (see Fig.6).

This factorization foregrounded common morphological entities, such as roots, schemes, with multiple conjugating elements (e.g., present, past, singular, dual, plural, masculine, feminine, etc) and agglutinations (proclitic and enclitic).

As depicted in Fig.6, the sub-cloud representing the root "**بعد**" **/baEada/** (i.e., *to go far*) is linked to its derivatives that are presented as molecules. On the other hand, Fig.7 displays the union of different sub-clouds (black balls stand for sub-cloud nuclei). The latter correspond to the roots in order to form

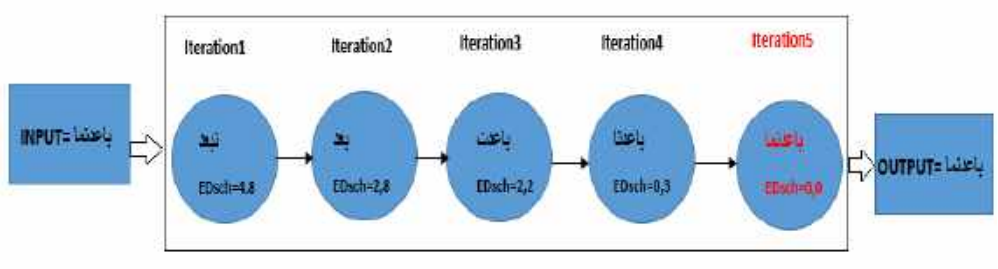



**Fig. 16.** A sample of successful recognition with SA algorithm

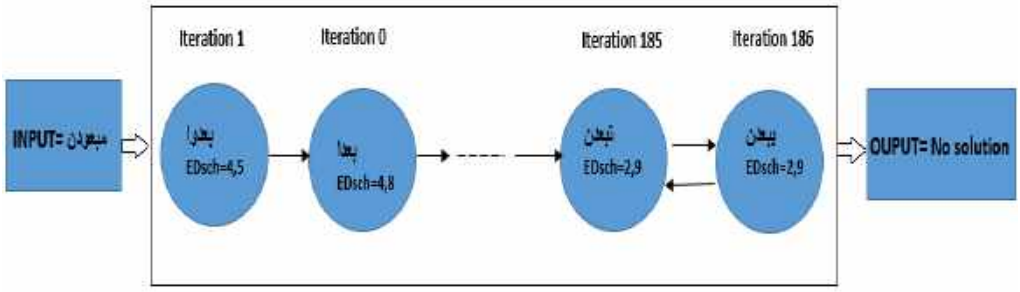



**Fig. 17.** A sample of misrecognition with SA algorithm

the entire molecular cloud. Thus, according to this structure, the notion of neighborhood appears two times:

On the one hand, it represents the root surrounded by all its derivatives obtained through scheme derivations forming a sub-cloud.

On the other hand, each root (a sub-cloud) is linked to a neighboring root which must be morphologically the nearest as shown in Fig.7.

This morpheme-based model of word structure enhances the learning process that characterizes most pattern recognition systems. The first step consists in preparing an initial cloud of morphologically related words by taking into account various verb conjugation rules in different tenses, with multiple persons, flexion, pronouns, and functions.

The second step focuses on incorporating some morphological instructions in order to link the generated words to sub-clouds. Finally, the first obtained cloud describes the initial solution space.

The latter is automatically stabilized in accordance with SA iterations. Note that our approach is not based on segmentation (segmentation free). It intends to integrate linguistic knowledge in both vocabulary structure and elastic comparison process.

### 7.2 CNN Training

As second step, based on EDsch algorithm, we tried to implement a convolutional neural network of Levenshtein distances that learns to calculate edit distances between Arabic words. Our CNN model is a set of an input and an output layer, along with numerous hidden layers, which are conventionally fully connected. The output of each layer represents the input of the next layer.

For hidden layers, they are composed of convolution layers, pooling layers followed by some fully connected layers and activation functions. The CNN model contains these different layers: (five convolution layers, two max_pooling layers, two dense layers, three dropout and one batch_normalisation layer).

In our approach, the obtained cloud is not only composed of classical nodes representing the words. It is henceforth, composed of linked neuron networks (CNN) each modeling a word. This way, we would not need an external algorithm to help

searching a word into the cloud, orchestrating the whole cloud. It is indeed about giving to each CNN-node the ability to give the search while calculating itself the needed elastic distance between any word and itself. Fig.8 shows the architecture of the implemented CNN_wi that's learns to calculate the edit distance between the word wi and any other word wj for the training corpus.

### 7.3 Recognition by Simulated Annealing

The Simulated Annealing (SA) can be considered as a metaheuristic search algorithm with a historical overview of successful applications to numerous optimization problems over a wide range of fields [15, 16]. Based on an initial candidate solution, SA iteratively seeks for a better neighbor solution to the problem than the current one. SA and other local search algorithms (e.g., Hill Climbing) are different.

Unlike the current candidate, SA is a global algorithm that can accept a worse solution during the iterative process. Therefore, the SA method has the ability to escape from local minima and converges towards global minima in order to reach its goal [17] (see algorithm in Fig. 9). By analogy to a physical system, an optimization problem can be described based on the parameters detailed in the table 1.

The performance of simulated annealing for specific problems depends on the choice of some parameters, such as the state space, temperature, the neighbor selection method, the probability transition function and the annealing schedule [17]. These choices have a great impact on the results. These parameters will be then adjusted according to the decision rule of our SA algorithm.

Thus, SA generally provides high quality solutions. It is a general, simple and flexible method since new constraints can be easily incorporated [18]. In a previous work [19], we proposed a SA algorithm based on Levenshtein comparative method in order to recognize decomposable Arabic words.

It is a classic edit distance calculation that does not distinguish between the root consonant or scheme and the conjugated letters that occur in prefixes and suffixes. Consequently, the classical distance causes confusion when searching the solution.

To bridge these gaps, we proposed in another work [20] an edit distance calculation (EDsch), inspired by some works [21, 22], adapted to our problem by embedding linguistic knowledge into the calculation. We have incorporated new costs in the editing operations, which emphasize the presence or absence of scheme letters.

These costs differ according to the location of the scheme letters. Looking at the example presented in table 2, the classical calculation regrettably gives the same edit distance (ED = 1) face to three extremely different cases:

The first case is a comparison between the word w1="**يبعد**" /**yaboEudu**/ (derived from the scheme "**فعل**" /**faEala**/) and w2="**يبتعد**" /yabotaEidu/ (derived according to the scheme "**افتعل**" /**AfotaEala**/). Note that w1 and w2 are very different and follow two different schemes. That is why a distance equal to 1 is not satisfying.

The second case is a comparison between the word w1 and the word w3="**مبعد**" /**muboEidN**/ (derived according to the scheme "**مفعل**" /**mufuEilN**/). Once again, w1 and w3 are very distant; they are not of the same scheme class. In fact, w1 is a verb while w3 is a noun. Hence ED=1 is not acceptable.

The third case is a comparison between w1 and the word w4="**يبعده**" /**yuboEiduhu**/ (derived according to the scheme "**فعل**"). Here, indeed w1 and w4 follow the same scheme and the same flections. The ED=1 is suitable, it is in fact simply due to an agglutinated letter at the end of the same word.

Instead of that, with the new Edit distance calculation, which highlights the pattern letters, the differences between words is emphasized. In fact:

- Comparing w1="**يبعد**" and w2="**يبتعد**" gives a weighty EDsch=2,9. It is indeed due to the use of two different schemes (two different scheme letters: one at the beginning and one in the middle). The EDsch is satisfying.
- Comparing w1="يبعد" and w3="مبعد" gives a notable EDsch=2,6 (< 2,9). It is again due to the use of two different schemes (only one different scheme letter at the beginning).
- Comparing w1="يبعد" and w4="يبعده" gives a notable EDsch=1,3 (<< 2,9). The distance is fairly small since, as already mentioned, w1 and w4 follow the same scheme.

It is obvious that the idea of the algorithm focuses on the scheme letters by giving them a greater weight than the other letters (of conjugation and of agglutination). Therefore, the problem of long stagnation in local wells, that we may encounter while using Levenshtein distance, is resolved due to the integration of new costs.

This way, in this work, we propose to embed a convolutional neural network in each node of the simulated annealing, so that any node can actually calculate edit distance and participate in finding the optimal solution in the word cloud.

The proposed algorithm is an SA algorithm based on adaptive Levenshtein comparative method (EDsch) and CNN classifiers. Each node of SA is a CNN able to determine the best (EDsch) with any word. Thus, the nodes of our simulated annealing are no longer sub-clouds of words; they are sub-clouds of CNN, which are ‘’linguistically” trained to see the distance between Arabic words.

## 8 Experimentations

In order to generate our training corpus reflecting our large vocabulary, we used Arabic word images derived from tri-consonantal healthy roots, extracted from APTI (Arabic Printed Text Image) database (see Fig. 10). Which contains a large vocabulary of decomposable and non-decomposable words written in 10 fonts, 10 sizes and 4 styles (plain, italic, bold and italic combined with bold).

This combination of fonts, styles and sizes ensures a high variability of images in the database. This generated a total number of word images of over 45 million [23].

As a first step, we built our molecular word cloud corpus from the APTI word images. We organized the images in subsets according to the roots. Each subset corresponds to a root and groups all the word images derived from this one forming a molecular cloud of words. The output of this part provides various and significant information that are helpful for our work.

In addition to the Arabic linguistic features (schemes, roots, conjugating elements, etc), our corpus includes information about the neighbors of each word. For example, Fig.11 describes a neighborhood sample between the following words; “**بعد**” /**baEada**/ with Id=1 which have two direct neighbors; the word “**تبعد**” /**taboEudu**/ with Id=0 and the word “**بعدن**” /baEadona/ with Id=2 and tow indirect neighbors; the word “**تبعدن**” /**taboEudona**/ with Id=3 and the word “**أبعدن**” /**>aboEadona**/ with Id=4.

Our input corpus is a set of structural primitives extracted from Arabic word images. The process of extraction is developed in the previous work [7]. We dealt with 3148 samples derived from 11 tri-consonant roots following several schemes and extended using a variety of agglutinative and inflectional features.

### 8.1 CNN Results

For the CNN configurations, we choose Tensorflow and Keras frameworks running on Python. The adopted convolutional layers used 64 filters each of the size (3*3), (ReLus) as activation function, dropout technique used after each layer to minimize the overfitting, Softmax Loss and accuracy metric used for model evaluation. Padding was set to zero when needed.

The number of epochs was set to 50 in the entire architecture. To evaluate our CNN classifiers, we selected randomly around 80% of word images for training while 20% were used for test.

Table 3 shows some CNNs scores. The CNN_0/0/6/10/4/0 is relative to the word “**تبعدن**” (derived from the root “**بعد**” which Id=0, following the pattern “**فعل**” which Id=0, in the present tense which Id=6, with the second person (Id=10), etc). It can calculate the edit distance between the word “**تبعدن**” and any other word with an accuracy equal to about 98,48%.

The CNN_0/0/5/11/5/0 corresponding to the word “**بعدن**” (derived from the root “**بعد**” which Id=0, following the pattern “**فعل**” which Id=0, in the past tense which Id=5, with the third person (Id=11), etc), can predict about 99,23% of correct edit distances.

The third example is the CNN_0/2/5/10/0/0 corresponding to the word “**باعدتما**“(derived from the root “**بعد**” which Id=0, following the pattern “**فاعل**” which Id=2, in the past tense which Id=5, with the third person (Id=11), etc) can predict about

88,55%of correct edit distance. Figures 12 and 13 illustrate, respectively, the following two metrics: accuracy and loss.

### 8.2 Simulated Annealing Results

Experimental tests show that the highest recognition rates (99.84%) are achieved with an initial temperature equal to 100 with stable cooling equal to 0.95. Furthermore, Fig.14 and Fig.15 demonstrates that the higher the cooling rate is, the more important the recognition rate is, with initial temperature fixed at 100.

Thus, a small cooling speed allows a good recognition rate (rapid cooling returns local solutions). Figure 16 shows a scenario of successful recognition of the word "**باعدتما**" (went back) by simulated annealing which achieves the optimal solution after only 5 iterations.

In contrast, Fig.17 presents a scenario of misrecognition of the word "**مبعودون**" (are banished) by simulated annealing algorithm which stagnates after 186 iterations and does not yield an optimal solution.

## 9 Conclusion

In this paper, we proposed an approach that takes advantage of neural methods and combinatorial optimization techniques to recognizing a large vocabulary of decomposable Arabic words considering the particular complexity and specific nature of the Arabic script. In a first step, we structured our vocabulary by integrating Arabic linguistic knowledge in order to generate a word cloud in the form of a molecular cloud.

This structure corresponds to the Arabic concept of building words according to their roots. The sub-clouds correspond to a neighborhood of words derived from the same root. In a second step, we used CNN techniques in the learning phase with a corpus of 1500 samples of derived from one root. Experiments were carried a morphological cloud of nodes structured on sub-clouds that corresponds to tri-consonantal Arabic roots.

For corpus, we have used structural primitive vectors extracted from word images including APTI samples. Furthermore, we incorporated linguistic knowledge in the recognition phase through the SA algorithm, especially in the edit distance calculation; we have precisely highlighted the scheme letters to help the SA algorithm to achieve.

Then, each node of SA is a CNN classifier allowing to determine the best edit distance (EDsch) between two words. Thus, the nodes of our simulated annealing are no longer molecular sub-clouds of words; they are molecular sub-clouds of CNN classifiers. Our future works will focus, on increase the dataset size by integrating more roots. Then, in order to reinforce our approach, we will try to test other deep learning techniques, like LSTM and Bi-LSTM.

## References

1. **Romain**, **R. (2019).** Contributions and perspectives on combinatorial optimization and machine learning for graph matching and classification. François Rabelais University, Tours, France.
2. **Kacem, A. K., Cheikh, I. B., Belaïd**, **A. (2014).** Collaborative combination of neuron-linguistic classifiers for large arabic word vocabulary recognition. International Journal of Pattern Recognition and Artificial Intelligence, Vol. 28, No. 1, pp. 1453001. DOI: 10.1142/S021800 1414530012.
3. **Cheikh, I. B., Allagui, I. (2015).** Planar Markovian approach for the recognition of a wide vocabulary of arabic decomposable words. 2015 13th International Conference on Document Analysis and Recognition ICDAR, IEEE, pp. 1031–1035. DOI: 10.1109/ ICDAR.2015.7333918.
4. **Kessentini, Y. (2009).** Modèles de Markov multi-flux pour la reconnaissance de l'écriture manuscrite multi-scripts. PhD thesis, University of Rouen.
5. **McClelland, J. L., Rumelhart, D. E. (1981).** An interactive activation model of context effects in letter perception: I. An account of basic findings. Psychological Review, Vol. 88, No. 5, pp. 375–407. DOI: 10.1037/0033-295X. 88.5.375.
6. **Belaïd, A., Choisy, C. (2008).** Human reading based strategies for off-line arabic word

recognition. In: Doermann, D., Jaeger, S. (eds) Arabic and Chinese Handwriting Recognition. SACH 2006. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg. Vol. 4768. DOI: 10.1007/978-3-540-78199-8_3.

7. **Cheikh, I. B., Zouaoui, Z. (2013).** HMM based classifier for the recognition of roots of a large canonical arabic vocabulary. In ICPRAM, pp. 244–252. DOI: 10.5220/0004335202440252.
8. **McCulloch, W. S., Pitts, W. (1943).** A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, Vol. 5, pp. 115–133. DOI: 10.1007/BF02478259.
9. **Benbakreti, S. (2019).** Développement d'un système de reconnaissance en ligne du manuscrit Arabe. Doctoral dissertation, Université de Sidi Bel Abbès-Djillali Liabes.
10. **Kanoun, S., Alimi, A. M., Lecourtier, Y. (2010).** Natural language morphology integration in off-line arabic optical text recognition. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 41, Vo. 2, pp. 579–590. DOI: 10.1109/TSMCB.2010.2072990.
11. **Khamekhem-Jemni, S., Kessentini, Y., Kanoun, S. (2019).** Out of vocabulary word detection and recovery in Arabic handwritten text recognition. Pattern Recognition, Vol. 93, pp. 507–520. DOI: 10.1016/j.patcog.2019.05.003
12. **Cheikhrouhou, A., Kessentini, Y., Kanoun, S. (2020).** Hybrid HMM/BLSTM system for multi-script keyword spotting in printed and handwritten documents with identification stage. Neural Computing and Applications, Vol. 32, pp. 9201–9215. DOI: 10.1007/s00521-019-04429-w.
13. **Ahmed, R., Gogate, M., Tahir, A., Dashtipour, K., Al-tamimi, B., Hawalah A., El-Affendi, M. A., Hussain, A. (2021).** Novel deep convolutional neural network-based contextual recognition of arabic handwritten scripts. Entropy, Vol. 23, No. 3, pp. 340. DOI: 10.3390/e23030340.
14. **Bello, I., Pham, H., Le, Q. V., Norouzi, M., Bengio, S. (2017).** Neural combinatorial optimization with reinforcement learning. DOI: 10.48550/arXiv.1611.09940.
15. **Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. (1983).** Optimization by simulated annealing. Science, Vol. 220, No. 4598, pp. 671–680. DOI: 10.1126/science.220.4598.671.
16. **Huang, K. Y., Hsieh, Y. H. (2011).** Very fast simulated annealing for pattern detection and seismic. IEEE International Geoscience and Remote Sensing Symposium, pp. 499–502. DOI: doi: 10.1109/IGARSS.2011.6049174.
17. **Mahdi, W., Seyyid, A. M., Ouali, M. (2017).** Performance analysis of simulated annealing cooling schedules in the context of dense image matching. Computación y Sistemas, Vol. 21, No. 3, pp. 493–501. DOI: /10.13053/cys-21-3-2553.
18. **Xinchao, Z. (2011).** Simulated annealing algorithm with adaptive neighborhood. School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China, Applied Soft Computing, Vol. 11, No. 2, pp. 1827–1836. DOI: 10.1016/j.asoc.2010.05.029.
19. **Zeineb, Z., Imen, B. C., Mohamed, J. (2018).** Combinatorial optimization approach for arabic word recognition. Proceedings of the 2018 International Conference on Artificial Intelligence and Pattern Recognition, pp. 15–20. DOI: 10.1145/3268866.326888.
20. **Zouaoui, Z., Ben-Cheikh, I., Jemni, M. (2019).** Combinatorial optimization approach for arabic word recognition based on adaptive simulated annealing. In: Nyström, I., Hernández Heredia, Y., Milián Núñez, V. (eds) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, CIARP 2019, Lecture Notes in Computer Science, Vol 11896. Springer, Cham. DOI: 10.1007/978-3-030-33904-3_45.
21. **Gueddah, H. (2017).** La correction orthographique des textes arabes: Contribution à la résolution d'ordonnancement et de l'insuffisance des lexiques. PhD thesis, University of Mohamed V, Rabat.
22. **Abdellah, Y., Lhoussain, A. S., Hicham, G., Mohamed, N. (2020).** Spelling correction for the arabic language space deletion errors.

Procedia Computer Science, Vol. 177, pp. 568–574. DOI: 10.1016/j.procs.2020.10.080.

23. **Slimane, F., Ingold, R., Hennebert, J. (2017).** Competition on multi-font and multi-size digitally represented arabic text. ICDAR, pp. 1466-1472. DOI: 10.1109/ICDAR.2017.239.
24. **Tu, Z., Zheng, S. Yuille, A. (2008).** Shape matching and registration by data-driven EM. Computer Vision and Image Understanding, Vol. 109, No. 3, pp. 290–304. DOI: 10.1016/j.cviu.2007.04.004.
25. **Yin, P. Y. (2000).** A tabu search approach to polygonal approximation of digital curves. International Journal of Pattern Recognition and Artificial Intelligence, Vol. 14, No. 2, pp. 243–255. DOI: 10.1142/S0218001400000167.

*Article received on 26/09/2022; accepted on 05/05/2024.*
** Corresponding author is Zeineb Zouaoui.*

ISSN 2007-9737

# Survey of Mobile Cloud Computing Security and Privacy Issues in Healthcare

Rabab M. Nabawy[1,2,*], Mohamed Hassan-Ibrahim[1], Mostafa Rabee-Kaseb[1]

[1] Fayoum University, Computer Science, Fayoum, Egypt

[2] October 6 University, Computer Science, Egypt

{rm2227, mhi11, mrk00}@fayoum.edu.eg

**Abstract.** Wireless positioning is regarded as an essential research direction across various domains. There are several wireless positioning algorithms available, with two-step positioning methods being the most significant. In order to increase positioning accuracy and efficiency, standard two-step positioning algorithms extract measurement data from the received signal, such as angles of arrival (AOA), times of arrival (TOA), and time differences of arrival (TDOA). The omnipresent healthcare system can benefit from the effective solutions offered by wireless sensor networks. A key element of the healthcare system of the future is the wireless sensor network. Cloud computing (CC), fog computing (FC), Internet of Things (IoT), and telehealthcare technologies are utilized in the healthcare industry to facilitate data sharing across diverse stakeholders. Healthcare data infringement might result from an unsafe healthcare method, giving hackers complete access to patient email addresses, messages, and reports. On the other hand, a secure method for healthcare 4.0 can raise stakeholder, patient, and caregiver satisfaction. These facts serve as the foundation for this study, which offers an extensive literature evaluation of several security issues and privacy-related healthcare concerns. Challenges and future research directions for achieving security and privacy in healthcare will be presented.

**Keywords.** Cloud computing, internet of things (IoT), geographical positioning system (GPS), security, privacy.

## 1 Introduction

The medical field offers a program "to keep people healthy," which helps patients survive [1]. Both improving the diagnostic process utilized in healthcare and developing the technologies employed by healthcare professionals can help achieve this goal.

There have been several developments in the healthcare field. 1.0 to 4.0 for medicine. Healthcare 1.0 permits doctors to maintain handwritten patient medical history records.

In contrast, these handwritten records were superseded by electronic records in Healthcare 2.0. Healthcare 3.0 uses wearable devices (W.D.s) to track patient medical history in real-time [2]. In the end, an electronic health record (EHR) system was created that enables patient data to be kept in a database repository that enables accessing it from anywhere in the world over the Internet.

Protecting patient privacy is one of the most significant part of making sure of data integrity. Healthcare 4.0 will save patient data in the centralized HER system, which continually monitors patients' health information and offers real-time services to patients [3].

Wearable technology and medical devices that are implanted in people can both be used to track their health. W.D.s are equipped with a variety of medical sensors, a procedure known as telehealthcare, to get the patient's blood pressure, heart rate, temperature, and glucose level remotely and keep them in the centralized HER [4].

Understanding the patient's behaviour is the major duty in order to deliver better remote patient care. IoT and telehealthcare can collaborate to manage illnesses more effectively [5].

Health monitoring, autonomous driving, smart grids, smart homes, intelligent transportation systems, smart devices, and mobile device (robot) placement are only a few of the many positioning services presented by the intelligent society [1-2]. WSN is growing in popularity across a wide range of sectors and is receiving greater attention in research as the Internet of Things (IoT) expands.

Furthermore, because the GPS method cannot meet requirements in an inside environment, the WSN methodology may make up for GPS's shortcomings and provide accurate localization services. Localization is achieved by determining and measuring the interspace between the mobile terminal and sensor nodes. Using distance measurements, such as received signal strength (RSS) [5, time of arrival (ToA) [3, time difference of arrival (TDoA) [4, angle of arrival (AoA) [6, etc., a range-based localization technique finds locations.

### 1.1 Scope

Numerous studies have been conducted to draw attention to the security and privacy concerns with healthcare 4[3]. The majority of these studies highlight privacy and security issues and how different healthcare sectors have addressed them. We have outlined the many Healthcare 4.0 application areas in the recommended evaluation and spoken about how integrating more technologies may benefit the healthcare sector. The confidentiality and privacy of the patient's medical data are guaranteed by this integration, which also improves diagnosis. Healthcare IoT presents both potential and constraints, as exemplified by Baker et al. [24]. Attacks against EHR systems were surveyed by Priya et al. [22], with a focus on known security flaws.

### 1.2 Contributions

The review presents in detail security and privacy concerns in healthcare. We highlight many open problems and challenges for privacy and security. The next section represents the main parts of this review:

– The background and the need for both security and Privacy in Healthcare 4.0.
– Merits and demerits of security and Privacy in Healthcare 4.0.
– Different techniques that are used to improve security and Privacy in Healthcare 4.0.
– The categorization of various security and privacy methods.

## 2 Background

### 2.1 Classification of Localization

Within this area, we presented indoor localization techniques for smartphones that were categorized based upon the different methods used by various studies [1]. They are categorized as Path Loss Prediction, Dead Reckoning, and Mapping. These approaches may employ one or more of the measurement strategies covered in the preceding section, and they may have made use of one or more smartphone sensors or components for position estimation.

### 2.2 Mapping

Fingerprinting, another name for mapping [2], is the process of creating a model of signal intensity. By comparing the detected signal intensities to the previously stored locations on the map, the approach estimates the position of the mobile device. The offline phase and the online phase are two stages.

### 2.3 Path-loss Prediction

A smartphone's estimated position may also be determined by figuring out how far away it is from three B.S. or A.P. using different signal propagation route loss models. A.P. may receive signals from a device that takes distinct routes and experiences absorption, deflection, or reflection.

### 2.4 Dead Reckoning

This method calculates an object's ultimate location by estimating its speed over a certain distance and time, and it uses a previously calculated position that was acquired by measuring some external references, such as a GPS, to

assess an object's present position [2-4]. Step event detection, heading direction estimate, step length computation, and position estimation are its four stages.

## 3 Wirelessly Networked Sensors in Healthcare

Wireless network sensors suggest sampling physiological, behavioural, cognitive, and physical processes that differ across individuals, buildings, and even bigger ones. Applications in healthcare based on sensory input create such massive sampling over spaces of different sizes [1]. Information is gathered from several dispersed sensors.

In addition, the sophistication of sensing has increased tremendously with the development of inexpensive, immature sensors; however, high-quality sensors are needed for home and personal use, as well as the creation of complex machine learning algorithms that allow complex conditions like stress, depression, and addiction to be extracted from sensory data. Lastly, the evolution of ubiquitous Internet connectivity has made it easier for caregivers to receive sensor data on time. We present a list of these technologies' uses in healthcare in the sections that follow.

## 4 Issues in Localization

When it comes to WSN localization, localization accuracy is the most important factor to consider since, without it, WSN is useless. A detailed description of a few more problems that significantly impact WSN may be found below.

### 4.1 Accuracy

WSN accuracy is crucial for the majority of localization applications [13]. The accuracy becomes higher in this instance by calculating the distance between two nodes using Euclidean measurement.

### 4.2 Energy Consumption

Given that WSNs have limited resources, energy consumption is a critical concern. WSN is made up of tiny sensors with changeable or limited-energy batteries that cause problems for the network when they run low [2]. The transfer of data uses more energy. Energy is needed even when the sensors are not in use.

### 4.3 Overhead

The costs of the sensor nodes include expenses for hardware, processing, and communication incurred by the sensor nodes are included in the overhead of localization techniques. A big, effective data structure is needed for localization in a 3D network [1].

### 4.4 Optimization

Every area of human endeavour and every computer science engineering research project involves some degree of optimization [4]. Biologically driven computational techniques, such as artificial bee colony (ABC), genetic algorithm (G.A.) [5], particle swarm optimization (PSO) [1], and differential evolution (D.E.) [2], are integrated into localization methods.

### 4.5 3D Deployment

Traditionally, localization has meant figuring out where the 2D WSN sensor node is. Nevertheless, it is used in 3D in different contexts, including environment monitoring, space exploration, surveillance, and undersea ecosystems [22].

### 4.6 Security and Privacy

In many applications, privacy and data security are considered the major concerns. The Internet of Things and related services are growing quickly, and this is one of the key drivers.

This may lead us to consider a number of issues, including the authentication protocol. It's also essential to examine current assaults and their defences. Thus, the identical behaviour is seen by the authentication point.

The majority of the studies concentrate on the authentication process between the tags and the reader, showcasing novel approaches to safeguard this communication versus numerous known attacks; similarly, the ownership transfer topic showcases novel approaches to enforce security during the actual tag transfer, protect the parts' privacy, and investigate the process' scalability to secondary markets.

In order to stop an attacker from pretending to be someone else or from violating the shared authentication procedure, security concerns have to be taken into serious consideration.

### 4.7 Security Measures

IoMT is now confronted with several security challenges, including inadequate security and privacy protocols as well as a lack of awareness and training. There are multiple categories that separate cryptographic and non-cryptographic solutions. Five tiers of security solutions will be offered in order to identify and avert threats.

Additionally, to protect patient privacy and lessen the harm caused by these known assaults. By providing physical flexibility and mobility, patients' health will be monitored in real-time.

#### 4.7.1 Non-technical Security Measures

Non-technical security measures ensure the securing of the medical information of the patients and also train the staff; training the medical and I.T. staff could be carried out in three different ways:

a. Raising Awareness,
b. Technical Training,
c. Raising the Education Level.

#### 4.7.2 Technical Security Measures

Using an end-to-end secure Internet of Medical Things system for the adoption of technical security solutions needs to be the first priority.

– Authentication and Certification Using several variables: A strong authentication and verification process has to be in place to safeguard IoMT systems against unwanted access. There is a vast array of biometric techniques, which are classified into physical and behavioural processes. For example, these include: • Physical biometric techniques, such as iris or retinal scans or facial recognition software.

– Cognitive, movement, speech, keyboard, mouse, and other behavioural biometric techniques, as well as signature recognition and other biometric techniques, are also used.

– Multi-Factor Authentication Techniques: By establishing authentication first, you can confirm the authenticity of both the source and the destination.

– Authorization Techniques: The minimal privilege has to be used for authorization.

– Techniques for Availability: To ensure constant data flow, servers must be maintained.

Honeypots: Systems that utilize them must be able to identify attackers along with their tools, goals, and techniques in order for them to be effective.

## 5 Taxonomy of Security and Privacy in Healthcare

The global taxonomy of security and privacy issues in Healthcare 4.0 concerns machine learning (ML), W.D., IoT, telehealthcare, policy, scheme, and network traffic-based security and privacy issues in healthcare. The proposed classification is shown in figure1.

### 5.1 Processing-based Schemes

The speedy growth of technology in the field of information and communication integrates a huge amount of data, which we call B.D.

This marvellous amount of data transmitted over the Internet, which is an open channel, and to manage security attacks, is a main challenging issue in processing-based techniques like CC, F.C., and Mobile edge computing (MEC) [15].


**Fig.1.** Proposed taxonomy of healthcare security

#### 5.1.1 Distributed Scenario

Users store data on their local storage so that only one user can access it. In case of hardware failure, it becomes difficult to recover data from the server. With a distributed system, such a problem can be easily solved, while other machines can process the request to avoid the request failure.

#### 5.1.2 Centralized Scenario

Nowadays, a significant volume of data is being produced as conventional systems cannot handle the vast amount of information. This is because of advances in technology. Therefore, effectively overseeing and regulating it can be difficult duties that can be accomplished with the help of a framework such as the cloud.

Zhou and colleagues [58] created a dynamic method for mining medical texts and extracting image features in the cloud healthcare system while ensuring privacy. It decreases the expense of computing and sending data while enhancing the security of input and output data.

#### 5.1.3 Decentralized Scenario

Creating a decentralized system is essential nowadays to build trust among all parties in a multiparty network [15]. The institution-driven approach emphasizes the sharing of information among different business organizations, such as various hospital organizations. In a setting where patients are in control, stakeholders have access to the patient's EHR data via APIs.

The evaluation of blockchain security was conducted by analyzing five elements: data aggregation, digital access regulations, data liquidity, immutability, and patient identity [16].

### 5.2 Machine Learning-based Schemes

The common use of cloud, mobile, online, and IoT technologies raises the risk of security breaches. ML picks up on cues from its surroundings and handles complicated scenarios with ease. It is noticed that the scientific method with the greatest degree of adaptability and finds usage in a widespread of fields, including image processing, security, autonomous driving, and network intrusion detection. Additionally, it may be applied to the prediction of illnesses using various datasets.

### 5.3 Wearable Device-based Schemes

Wearable devices (WDs) are gadgets that individuals wear to monitor health data like sleep patterns, heartbeat, blood pressure, body temperature sensors, and exercise data. It aids in enhancing the patient's quality of life. Data produced by W.D. is transmitted to the patient's Mobile using various communication channels like blue-tooth and zig-bee, as shown in Figure 2.

## 6 Related Work

Localization is the act of identifying the geographic position of a user or device. The process may involve identifying a user and tracking their movement, or it may not. Therefore, the proposal could identify areas of focus for isolation and gradual economic resurgence.

Experts recommended the use of a sensor network that utilizes lightweight remote sensor hubs equipped with IoT technology to communicate information through distributed computing. Therefore, they adhere to privacy and security protocols to access the patient care

system and maintain the confidentiality of information. In e-health systems, experts studied combining IoT and cloud computing, identifying and outlining relevant hurdles, prospects, and constraints.

They provided evidence of the crucial nature of e-Health platform security and suggested a validation approach for obtaining e-Health frameworks with consistent deployment. Current surveys were examined in order to analyze the most efficient methods for protecting patient medical records.

Numerous research studies and experiments on mobile phone localization are currently being conducted in the healthcare community. It has the ability to monitor every alteration in the area. Examined were recently published papers showcasing innovative techniques for localization. Their attention to detail, research field, method of categorization, and methods of comparison were observed and understood.

In the article Akbarzadeh et al. (2021) [14], simple positioning methods without an emphasis on precision are introduced. As a result, BLE-based solutions enable lower installation costs and power consumption. However, this depends on the essential neighbourhood commercial services, like iBeacon. When anticipated systems are supposed to work on par with proximity-based ones, their functionality is limited.

The required accuracy for a restricted set of tests proposed that the suggested system outperforms a remote solution in terms of performance. In addition, little tweaks that reduce power consumption can be applied to systems like Wi-Fi. Unfortunately, limitations in the available technology make it impractical to perform a full power consumption measurement. Two cells are being used in the test.

The recommended fix enables a multi-year battery life extension for the beacons. This study's theoretical contribution, which suggests a mobile app for effective hospital administration, seeks to improve the body of existing research. Consequently, in order to improve hospital amenity management, the recently developed platform analyzes data on structures, their clients, and the period of their visits. In addition, by keeping an eye on available space and automatically modifying lighting and temperature to maximize energy

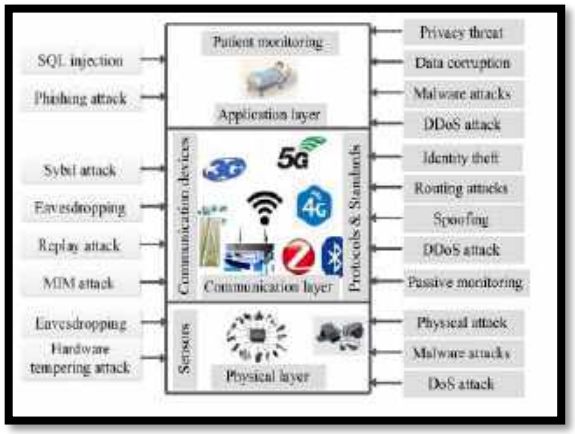



**Fig.2.** Possible attacks on a different layer

efficiency, the present system's characteristics may be improved.

Nasr et al. (2021) [15] consider smart healthcare guarantees the effectiveness of healthcare when compared to other options. Providing healthcare facilities with affordable services through a straightforward and safe process efficient and affordable system for monitoring health.

The study looked at cutting-edge wearables and smartphones that might monitor basic health information, use machine learning to identify COVID-19, diabetes, and heart disease, and provide support to older people living in their homes.

This review emphasizes the importance of software integration frameworks in creating smart healthcare systems. We have evaluated the benefits and drawbacks of several options. We also spoke about the primary issues that contemporary healthcare systems need to resolve in order to create efficient support models. Suggested new lines of inquiry to improve the current healthcare infrastructure.

Technology may help medical professionals by bringing new organizational structures, even if they cannot replace the medical system entirely. By working together on a common platform, researchers and medical professionals may create assistance solutions.

The article Soares Silva et al. (2021) [17] examined how cities in a heavily affected state in Brazil were

cut off by analyzing networks formed by mobile phone location data. By using mobility thresholds, clusters are formed to analyze the behaviour of 192 cities before and after the enactment of social distancing regulations. Thus, it was possible to witness the towns with varying degrees of connectivity.

The weighted flow-based clustering situations are a part of what this method offers; they could assist policymakers in determining if they should isolate a lone city or a cluster of cities with border regulations. One of the study's additional enhancements is cluster risk assessment. It involves prioritizing factors from most crucial to least crucial based on selected risk measures and specific circumstances.

When economic recovery measures are implemented, isolated clusters without instances may receive preferential treatment. As a result, economically sustainable clusters can be established to maintain both strong and weak connections between cities.

Our proposal aims to collect data for policymakers to determine areas for isolation and economic recovery, taking in consideration epidemiologists' views. Epidemiologists must determine isolation protocols, criteria for lockdowns, and reopening regions based on the timing of infections and the availability of healthcare facilities. Future research projects might include modifying clustering techniques to analyze small-world effects in weighted directed graphs.

In this case, different groups can be distinguished and analyzed without requiring specific flow minimums to alter the original graph structure. Additionally, the risk assessment could involve data regarding the hospital's specific location within each city or region. Furthermore, this article analyzes various states in Brazil.

This article Horng and Chen (2021) [18] suggested a fuzzy theory-driven adaptive threshold algorithm to detect the fall. If a fall happens, Beacon quickly detects where the caregiver is and sends information about the fall's location and identity via Wi-Fi to the server. Notifications are then sent to the caregiver's mobile app through the server. The system allows current healthcare providers to access additional real-time data and feedback, resulting in enhanced service quality and efficiency, leading to reduced human costs and resource usage.

The research Rhayem et al. (2021) [19] was conducted and put into practice with the aim of monitoring healthcare, which can be keept track of elderly individuals with chronic illnesses, pregnant women, and individuals with disabilities. These situations are connected to the ongoing changes in limitations and demands for the deployment of MCOs (time, place, condition). Additional information from the patient's medical record should be taken into consideration in order to guarantee a more accurate and appropriate treatment.

This article investigates the problem and presents a contextually aware system (IoT Medicare system) utilizing semantics to monitor patients with MCOs. The main domain ontology (HealthIoT-O) is applied to define the meanings of different MCOs and their data in the system.

The article McConville et al. (2021) [20] presented Vesta, a cutting-edge, reasonably priced digital health platform designed to track patients' whereabouts, health, and overall well-being in detail.

The platform's "smart home in a box" concept was deliberately designed to be more cost-effective than other systems, which added to its widespread appeal as an approachable tool for data collecting. It described how Vesta combined important data with the house by gathering accelerometer data from a wrist-worn wearable device with RSSI readings from four sites. The analytics system examines activity, interior positioning, and sleeping patterns in the smart home to transform raw sensor data into meaningful knowledg by using data science and machine learning.

In order to evaluate Vesta's efficacy, two case studies were looked at: one included patients having heart valve surgery, and the other featured a group of twenty patients. Three distinct periods of patient behaviour and activity at home—one prior to surgery and two following—could be seen on the platform.

The results were confirmed using a tailored survey including questions on physical activity and sleep quality, as well as standardized clinical Patient-Reported Outcome Measures (PROMs). The suggested platform has the potential to

enhance the clinical evaluations by incorporating quantitative health data from popular home monitoring devices into digital health research.

In Amine et al. (2021) [21], this system's primary flaw is that it may be used for more than just patient monitoring; machine learning can be suggested for classifying data. Because it may assist diabetic patients, their families, physicians, and medical researchers in making decisions about their care based on large data sets, predictive analytics is particularly important for diabetes patients. This research discusses predictive analytics utilizing four distinct machine-learning algorithms and outlines a new approach for monitoring diabetes patients.

Singha and Chatterjee (2021) [22] put forth a privacy-preserving paradigm in which various entities communicate with one another inside this architecture to carry out certain functions, including gathering, aggregating, storing, and analyzing data while maintaining data privacy. Below is a description of each entity:

- Smart User: People who are smart may also be healthy. Patients in their later years, those in critical condition, hospital patients, physicians, nurses, lab technicians, etc. In order to detect biosignals, people have had bio-sensors implanted in their bodies. Subsequently, they are combined in Internet of Things devices and protected cloud storage.
- Edge Gateway: This smart gadget is in charge of locally processing little data collected from the community of smart patients. Smart equipment such as oxygen delivery systems and cardiac pumps are controlled by means of this local diagnosis of data. In addition, it encrypts the data, puts it in cloud storage for more research, and allows physicians to access it for patient care.
- Database Manager: This manager is in charge of processing queries and storing data. The encryption key from the key generator will be searched using the processed query.
- Server Edge. This will handle all data resources from various intelligent community members and group them into various clusters for encrypted analysis.

Sivan R and Zukarnain (2021) [23] suggested a method for encrypting and securing identity-based data exchange. Many options were provided to secure data stored in cloud-based e-health systems, making use of existing PKE, IBE, IBBE, and ABE methods. To safeguard data security, an appropriate security solution needs to be created and kept up to date. Additionally, they emphasized a thorough analysis of current cloud-based e-health systems that use both non-cryptographic and cryptographic methods in order to safeguard the security and privacy of digital data.

Kondaka and Thenmozhi (2021) [24] suggested the method, known as iCloud-assisted Intensive Deep Learning (iCAIDL), which unites two potent techniques into one cohesive system. Our initial approach involves utilizing the Internet of Things in conjunction with the Cloud Paradigm to facilitate communication and interaction between our suggested smart device and distant servers.

The cloud paradigm offers sufficient storage capacity contingent on data needs and the number of patients or senior citizens. The second tactic involves utilizing machine learning techniques in conjunction with the Cloud IoT paradigm to anticipate challenging circumstances for patients. This allows medical professionals or other caregivers to take appropriate action in response to the event.

The suggested iCAIDL logic makes it possible for the end user to identify patient or elderly people's health records in a more thorough way without worrying about human error. Additionally, all metrics are fully transparent to the doctor and caregivers' end for additional norm verification.

Viswanathan (2021) [25] proposes using an energy-harvesting healthcare IoT system to improve download speeds and local computation without compromising data security, IoT efficiency, or high-level computing models. The system utilizes a data protection allocation scheme that is based on RL. This plan assesses the level of anonymity, energy consumption, and measurement delay to establish the download protocol. The RL-powered offloading strategy employs a transition learning technique, an acknowledged radio channel model, and Dyna architecture to accelerate learning for complex healthcare systems in DIT.

In Aitzaouiat and Latif (2021) [26], in order to convert between different nodes, such as the Hospital Information System (HIS) HTTP/TLS

protocol and the CoAP/DTLS protocol, the suggested approach uses a model as an integrated IoT/WEB proxy security.

The suggested method carried out better than existing approaches for many reasons: It is a secure prototype implementation that fulfils the following requirements: a) it makes use of a hierarchical collection of machine learning and prediction algorithms; b) it is user-friendly, open-source, and interoperable; c) it uses the correlation criteria to achieve a greater accuracy rate.

AlZubi and Al-Maitah (2021) [27] proposed that cognitive machine learning facilitates safe healthcare data exchange by supporting the attack detection architecture. Cloud storage of the gathered data will be facilitated by the Healthcare Cyber-Physical Systems. Cyber-attack behaviour is predicted by machine learning algorithms, and analyzing this data might help medical professionals make decisions.

The basic foundation of the suggested method is a patient-centric design that protects data on a reliable device, such as the end users' smartphones and gives them a choice over data-sharing access. According to experimental data, the proposed model outperforms other current models with an attack prediction ratio of 96.5%, an accuracy ratio of 98.2%, an efficiency ratio of 97.8%, a shorter latency of 21.3%, and a communication cost of 18.9%.

Through the integration of natural language capabilities, extensibility tools, compliance constructs, and integrated medical intelligence, Kumar Prasad et al. (2021) [28] have made it possible for healthcare organizations such as payers, providers, pharmacies, HMOs, and telehealth to give people access to reliable and pertinent healthcare information and services.

Bots with anomaly detection systems have further advantages over healthcare systems. This study employs the C2B-SCHMS framework, which uses machine learning isolation graph techniques to find any anomalies in the dataset. Because of this, cloud computing for medical services will be reliable.

Farrokhi and Farahbakhsh (2021) [29] gives a worldwide overview of the real issues with smart fitness and their fixes. In order to achieve this, three areas of smart fitness have been examined: artificial intelligence, social IoT, and IoT-based solutions (such as fitness trackers, fitness applications, and movement analysis). In terms of practical solutions and designs pertaining to IoT-based solutions, an AI-based recommendation accompanied by a comprehensive monitoring system remains challenging. Artificial intelligence (AI) in smart fitness is used for tasks including extracting exercise aspects, forecasting diets and workouts, preventing injuries, and overtraining.

Additionally, social IoT is a crucial component of the smart fitness space that may enhance knowledge of smart fitness through the sharing of experiences with different solutions, sensor kinds, techniques, and individuals with varying cultural backgrounds, socioeconomic statuses, and even personal habits. Smart Fitness may function as a coach's squad of assistants, assisting in improved decision-making.

It should be able to handle training information history management, workout and diet prediction monitoring, user authorization and identification, and fitness big data analysis in order to be a comprehensive and perfect smart fitness solution. There is still a lot of room for improvement when it comes to smart fitness.

In order to calculate the distance between a beacon node in an indoor setting and a mobile node carried by Alzheimer's patients, the study by Munadhil and Gharghan (2021) [30] proposed two route loss models. Because of the building's restricted space, the tested distance could only be 26 meters. Because of its robust characteristics and features compared to other wireless technologies, the ZigBee wireless protocol was adopted.

In trials with volunteers to verify the algorithm's accuracy, patients with high fall risk, such as those with diabetes, stroke, physical disabilities, and osteoporosis, are recommended. The ageing population is taken into account in this study, which calls for more investigation.

Yi and Feng (2021) [31] created and demonstrated a method for gathering online information on lower-limb kinematics and kinetics. Specifically, continuous prediction of movement is carried out by combining EMG and IMU signals, leading to the development of a real-time capable inverse dynamic model.

This method offers a thorough prediction of kinematics and kinetics in a specific motion mode,

addressing issues with PR-based motion intent recognition methods. Furthermore, by taking into account the impact of EMD, a forecast is made prior to actual movement in order to address the algorithm's computational time. Therefore, this technique offers fundamental information on walking patterns for smart healthcare applications, such as remote diagnosis and monitoring of diseases.

Hamad and Dawod (2022) [31] focus on the primary implementation framework for Elastic Mobile Cloud Computing (EMCC) solutions, covering cloud setup, program dissemination, module extraction, module transmission, program execution, and outcome. Then, the article examines the primary data security concerns regarding users' mobile devices that were emphasized by EMCC: data stream interception and privacy deficiencies. This implies that utilizing risk management can lessen security risks linked to implementing EMCC programs. Assessing the risks related to each distribution method is crucial in order to accomplish risk management. They created a method to quantify risk for the EMCC modular task, enabling them to effectively overcome the primary challenge of the approach.

Kiani and Shahid (2022) [32] highlight potential causes of data leaks, including in-law disputes, subpar equipment, ignorance, and the absence of specialized local law enforcement. This article addresses IoHT device compliance difficulties with relation to healthcare data privacy and protection legislation, bringing attention to the rising need for a proper regulatory framework. Additionally, the research gives some solutions for enhancing the secrecy and security of IoHT implementation.

To meet the needs for mutual authentication and protect user anonymity, Nag and Chandrakar (2023) [33] developed a remote user authentication system based on smart cards. An automated validation Internet Security Protocol Application (AVISPA) has demonstrated through simulation that the proposed system can fend against both passive and aggressive physical threats. According to an informal security evaluation, the developed scheme is resistant to a variety of assaults; compared to analogous existing protocols, the recommended protocol exhibits more security characteristics and is more complicated in calculation cost, execution time, and communication cost.

In the apper Thatikonda and Padthe (2023) [34], there are two categories in the suggested method. The initial part of the paper explores the technique of data formulation modification to understand data correlation and assess variables using trained data. It helps to achieve both the development of data scale and data minimization. By using the subset selection to show the model's fitness based on the data, the selection feature is employed in the second section to validate the model. Additional examples of various Android applications are required in order to analyze the framework with respect to metrics such as data accuracy and F-measure. This work focuses on Chi-square, Gain Ratio, information gain, logistic regression analysis, One R, and PCA since feature selection is thought to be a problem.

Prabha and Kanagasabapathi (2023) [35] employ capacity tools and layered, private, information-nature-driven encryption approaches like MES, which use safe health information exchange. Comparative results show that, in the Cloud Environment (from numerous execution factors), this strategy works better than other well-liked strategies. The ensuing list includes some likely obstacles and outcomes of the proposed project.

In the work Nyangaresi (2023) [36], wireless body area networks have been used to gather and send patient health data to hospital medical specialists for review from a distance. Since the transmission takes place over the open Internet, this procedure opens the door to several attacks. They examine the recent history in order to make this situation better. It has been demonstrated, therefore, that a large number of these techniques have significant communication and computation costs.

Furthermore, the majority of these existing protocols are insecure due to inherent flaws. This work presents the development of a three-factor security mechanism that combines passwords, smart cards, and biometric data to overcome these issues. The ROR model and BAN logic were used to provide formal security, and the outcomes showed it was safe.

## 7 Conclusion

Performance metrics and security and privacy regulations must be adhered to by digital services. Hence, building resilience to cyberattacks is crucial for improving the filtering skills of users of applications. Smart healthcare provides a cost-effective health monitoring system that is portable, secure, and efficient.

This enables people to receive high-quality medical treatment at a more affordable price compared to the rates of hospitals and nursing facilities. This study briefly looked into wearables and mobile phones for using machine learning to monitor vital signs. Various systems are incorporated within software integration frameworks. Moreover, we discussed the main challenges associated with the latest smart healthcare systems, which are the main obstacles in developing working prototypes.

Future research should explore specific strategies to further enhance the safety of the healthcare system. Even though technology cannot fully take over the healthcare system, it can help alleviate some of the workload for healthcare professionals by providing certain services and systems. Collaboration between scientists and medical professionals can lead to the development of assistive technologies that offer a foundation.

## 8 Challenges and Future Research Directions

Even if several assistive frameworks that highlight smart healthcare have been enhanced with the help of contemporary technologies, certain issues must be resolved to highlight a scalable, safe, conveniently accessible, and effective healthcare system. Combining data from various sensors is a significant obstacle to deploying wearable technology, such as smartphones, in the context of smart healthcare.

For health monitoring applications, it is crucial to transform the signals from diverse sensors attached to patients into a useful format because the numerous sensors provide several kinds of data. Future research can look at a number of data fusion approaches for combining information from multisensory devices in order to provide simplified signals that increase dependability and reduce the amount of bandwidth needed for connection with the cloud layer.

Different security protocols, like Blockchain, will retain more security and privacy and are advised to provide safe data exchange between users. Therefore, it is critical that these three factors be taken into consideration when developing AAL systems and that future research with older persons adhere to user-based testing and improvement. Last but not least, privacy and security issues pertaining to health-related data are crucial. Therefore, these elements need to be incorporated early on into the design of smart healthcare frameworks, making use of the most recent ones.

## References

1. **Gupta, R., Tanwar, S., Tyagi, S., Kumar, N., Obaidat, M. S., Sadoun, B. (2019).** Habits: blockchain-based telesurgery framework for healthcare 4.0. 2019 international conference on computer, information and telecommunication systems (CITS) IEEE**.** pp. 1–5. DOI: doi: 10.1109/ CITS.2019.8862127.
2. **Hathaliya, J. J., Tanwar, S., Tyagi, S., Kumar, N. (2019).** Securing electronics healthcare records in healthcare 4.0: A biometric-based approach. Computers & Electrical Engineering, Vol. 76, pp. 398–410. DOI: 10.1016/j.compeleceng.2019. 04.017.
3. **Coventry, L., Branley, D. (2018**). Cybersecurity in healthcare: A narrative review of trends, threats and ways forward. Maturitas, Vol. 113, pp. 48–52. DOI: 10.1016/j.maturitas. 2018.04.008.
4. **Shankar, K., Lakshmanaprabu, S. K., Khanna, A., Tanwar, S., Rodrigues, J. J., Roy, N. R. (2019).** Alzheimer detection using Group Grey Wolf Optimization based features with convolutional classifier. Computers & Electrical Engineering, Vol. 77, pp. 230–243. DOI: 10.1016/j.compeleceng. 2019.06.001.
5. **Gupta, R., Tanwar, S., Tyagi, S., Kumar, N. (2019).** Tactile-internet-based telesurgery system for healthcare 4.0: An architecture, research challenges, and future directions.

IEEE network, Vol. 33, No. 6, pp. 22–29. DOI: 10.1109/MNET.001. 1900063.

6. **Kumari, A., Tanwar, S., Tyagi, S., Kumar, N., Maasberg, M., Choo, K. K. R. (2018).** Multimedia big data computing and internet of things applications: A taxonomy and process model. Journal of Network and Computer Applications, Vol. 124, pp. 169–195. DOI: 10.1016/j.jnca.2018. 09.014.
7. **Mannay, K., Benhadjyoussef, N., Machhout, M., Urena, J. (2016).** Location and positioning systems: Performance and comparison. 2016 4th International Conference on Control Engineering & Information Technology, IEEE, pp. 1–6. DOI: 10.11 09/CEIT.2016.7929105.
8. **Rhayem, A., Mhiri, M. B. A., Drira, K., Tazi, S., Gargouri, F. (2021).** A semantic-enabled and context-aware monitoring system for the internet of medical things. Expert Systems, Vol. 38, No. 2, p. e12629. DOI: 10.1111/exsy. 12629.
9. **Shuaieb, W., Oguntala, G., AlAbdullah, A., Obeidat, H., Asif, R., Abd-Alhameed, R. A., Kara-Zaïtri, C. (2020).** RFID RSS fingerprinting system for wearable human activity recognition. Future Internet, Vol. 12, No. 2, p. 33. DOI: 10.3390/fi12 020033.
10. **Ghayvat, H., Awais, M., Gope, P., Pandya, S., Majumdar, S. (2021).** Recognizing suspect and predicting the spread of contagion based on mobile phone location data (counteract): a system of identifying covid-19 infectious and hazardous sites, detecting disease outbreaks based on the internet of things, edge computing, and artificial intelligence. Sustainable Cities and Society, Vol. 69, p. 102798. DOI: 10.1016/j.scs.2021.102798.
11. **Mohammad, G. B., Shitharth, S. (2021).** WITHDRAWN: Wireless sensor network and IoT based systems for healthcare application. DOI: 10. 1016/j.matpr.2020.11.801.
12. **McConville, R., Archer, G., Craddock, I., Kozłowski, M., Piechocki, R., Pope, J., Santos-Rodriguez, R. (2021).** Vesta: A digital health analytics platform for a smart home in a box. Future Generation Computer Systems, Vol. 114, pp. 106–119. DOI: 10.1016/j.future. 2020.07.046.
13. **Yi, C., Jiang, F., Bhuiyan, M. Z. A., Yang, C., Gao, X., Guo, H., Su, S. (2021).** Smart healthcare-oriented online prediction of lower-limb kinematics and kinetics based on data-driven neural signal decoding. Future Generation Computer Systems, Vol. 114, pp. 96–105. DOI: 10.1016/j.future.20 20.06.015.
14. **Akbarzadeh, O., Baradaran, M., Khosravi, M. R. (2021).** IoT-Based smart management of healthcare services in hospital buildings during COVID-19 and future pandemics. Wireless Communications and Mobile Computing, Vol. 2021, No. 1, p. 5533161. DOI: 10.1155/2021/ 5533161.
15. **Nasr, M., Islam, M. M., Shehata, S., Karray, F., Quintana, Y. (2021).** Smart healthcare in the age of AI: recent advances, challenges, and future prospects. IEEE Access, Vol. 9, pp. 145248–145270. DOI: 10.1109/ACCESS. 2021.3118960.
16. **Parvathy, V. S., Pothiraj, S., Sampson, J. (2021):** Automated internet of medical things (IoMT) based healthcare monitoring system. In: Hassanien A.E., Khamparia A., Gupta D., Shankar K., Slowik A. (eds) Cognitive Internet of Medical Things for Smart Healthcare, Studies in Systems, Decision and Control, Springer, Vol. 311, pp. 117, DOI: 10.1007/ 978-3-030-55833-8_7.
17. **Silva, J. C. S., de-Lima-Silva, D. F., Neto, A. D. S. D., Ferraz, A., Melo, J. L., Júnior, N. R. F., de-Almeida-Filho, A. T. (2021).** A city cluster risk-based approach for Sars-CoV-2 and isolation barriers based on anonymized mobile phone users' location data. Sustainable cities and society, Vol. 65, No. 102574. pp. 2210–6707, DOI: 10.1016/j. scs.2020.102574.
18. **Horng, G. J., Chen, K. H. (2021).** The smart fall detection mechanism for healthcare under free-living conditions. Wireless Personal Communications, Vol. 118, No. 1, pp. 715–753. DOI: 10.1007/s11277-020-08040-4.
19. **Rhayem, A., Mhiri, M. B. A., Drira, K., Tazi, S., Gargouri, F. (2021).** A semantic-enabled and context-aware monitoring system for the internet of medical things. Expert Systems, Vol. 38, No. 2, p. e12629. DOI: 10.1111/exsy. 12629.

20. **McConville, R., Archer, G., Craddock, I., Kozłowski, M., Piechocki, R., Pope, J., Santos-Rodriguez, R. (2021).** Vesta: A digital health analytics platform for a smart home in a box. Future Generation Computer Systems, Vol. 114, pp. 106–119. DOI: 10.1016/j.future. 2020.07.046.
21. **Rghioui, A., Lloret, J., Sendra, S., Oumnad, A. (2020).** A smart architecture for diabetic patient monitoring using machine learning algorithms. Healthcare, Vol. 8, No. 3, p. 348. DOI: 10.3390/ healthcare8030348.
22. **Singh, A., Chatterjee, K. (2021).** Securing smart healthcare system with edge computing. Computers & Security, Vol. 108, p. 102353. DOI: 10.1016/j. cose.2021.102353.
23. **Sivan, R., Zukarnain, Z. A. (2021**). Security and privacy in cloud-based e-health system. Symmetry, Vol. 13, No. 5, pp. 742. DOI: 10.33 90/sym13050742.
24. **Kondaka, L. S., Thenmozhi, M., Vijayakumar, K., Kohli, R. (2022).** An intensive healthcare monitoring paradigm by using IoT based machine learning strategies. Multimedia Tools and Applications, Vol. 81, No. 26, pp. 36891–36905. DOI: 10.1007/s110 42-021-11111-8.
25. **Aitzaouiat, C. E., Latif, A., Benslimane, A., Chin, H. H. (2022).** Machine learning based prediction and modeling in healthcare secured internet of things. Mobile Networks and Applications, Vol. 27, No. 1, pp. 84–95. DOI: 10.1007/s11036-020-01711-3.
26. **AlZubi, A. A., Al-Maitah, M., Alarifi, A. (2021).** Cyber-attack detection in healthcare using cyber-physical system and machine learning techniques. Soft Computing, Vol. 25, No. 18, pp. 12319–12332. DOI: 10.1007/ s00500-021-05926-8.
27. **Farrokhi, A., Farahbakhsh, R., Rezazadeh, J., Minerva, R. (2021).** Application of internet of things and artificial intelligence for smart fitness: A survey. Computer Networks, Vol. 189, p. 107859. DOI: 10.1016/j.comnet.2021. 107859.
28. **Munadhil, Z., Gharghan, S. K., Mutlag, A. H. (2021).** Distance estimation-based PSO between patient with Alzheimer's disease and beacon node in wireless sensor networks. Arabian Journal for Science and Engineering, Vol. 46, No. 10, pp. 9345–9362. DOI: 10.1007/ s13369-020-05283-y.
29. **Wang, Z., Rho, S., Yang, C., Jiang, F., Ding, Z., Yi, C., Wei, B. (2021).** Active loading control design for a wearable exoskeleton with a bowden cable for transmission. Actuators, Vol. 10, No. 6, pp. 108. DOI: 10.3390/act10 060108.
30. **Shahid, J., Ahmad, R., Kiani, A. K., Ahmad, T., Saeed, S., Almuhaideb, A. M. (2022).** Data protection and privacy of the internet of healthcare things (IoHTs). Applied Sciences, Vol. 12, No. 4, p. 1927. DOI: 10.3390/app120 41927.
31. **Nag, P., Chandrakar, P., Chandrakar, K. (2023).** An improved two-factor authentication scheme for healthcare system. Procedia Computer Science, Vol. 218, pp. 1079–1090. DOI: 10.1016/j.procs. 2023.01.087.
32. **Thatikonda, R., Padthe, A., Vaddadi, S. A., Arnepalli, P. R. R. (2023).** Effective secure data agreement approach-based cloud storage for a healthcare organization. International Journal of Smart Sensor and Adhoc Network, Vol. 3, No. 4.
33. **Sathya-Prabha, R., Kanagasabapathi, K., Sajeeth, K., Aishwarya, M. (2023).** Health information sharing in cloud environment using modular encryption standard. Recent Developments in Electronics and Communication Systems IOS Press, pp. 64–70. DOI: 10.3233/ATDE 221238.
34. **Nyangaresi, V. O. (2023).** Privacy preserving three-factor authentication protocol for secure message forwarding in wireless body area networks. Ad Hoc Networks, Vol. 142, pp. 1570–8705. DOI: 10.1016/j. adhoc.103117.

*Article received on 29/04/2023; accepted on 06/05/2024.*
**Corresponding author is Rabab M. Nabawy.*

ISSN 2007-9737

# Offensive Language and Hate Speech Detection Using Transformers and Ensemble Learning Approaches

Billel Aklouche[1,2,*], Yousra Bazine[2], Zoumrouda Ghalia-Bououchma[2]

[1] University of Constantine 2, Abdelhamid Mehri, LIRE Laboratory, Algeria

[2] University of Constantine 2, Abdelhamid Mehri, TLSI Department, Algeria

{billel.aklouche, yousra.bazine, zoumrouda.bououchma}@univ-constantine2.dz

**Abstract.** Social networks play a vital role in facilitating communication and information sharing. However, these platforms are also witnessing a growing prevalence of hate content, which can pose a major threat to individuals and entire communities. In this paper, we propose a new method that addresses the problem of offensive language and hate speech detection using seven transformer models, including BERT, and six ensemble learning strategies (Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging). Specifically, a fine-tuning process is run for each pre-trained model on hate speech detection downstream task. Subsequently, various ensemble learning techniques are applied by combining the predictions of individual models in order to improve overall performance. Extensive experiments have been conducted on the publicly available Davidson-dataset to assess the performance of our proposed method. Evaluation demonstrates promising results in terms of various evaluation metrics, outperforming competitive state-of-the-art baselines.

**Keywords.** Hate speech detection, offensive language, transformers, fine-tuning, ensemble learning, social media.

## 1 Introduction

Hate speech is a form of expression that aims to offend, hurt or discriminate against a person or a particular group. Indeed, it can have negative impacts on both the mental and physical health of victims [4]. According to the United Nations, it even affects the democratic values, social stability and peace. Therefore, detecting and preventing online hate speech is a crucial task for ensuring a safe and respectful environment for everyone. Social media platforms such as Twitter and Facebook are among the most common sources of hate speech, as they allow users to express their opinions and emotions freely and anonymously.

In fact, several suspects in recent hate-driven terrorist attacks had an extensive history of posting hateful content [11]. This emphasizes the crucial importance of detecting this kind of speech, particularly on social media. However, manually monitoring and moderating such a huge volume of user generated content is impractical, costly and time-consuming.

Therefore, there is a significant need for automated methods that can accurately and efficiently identify and classify hate speech within textual content on social media platforms [9]. In this paper, we propose a new method for automatic hate speech detection that leverages the performance of transformer-based models and ensembling techniques.

Indeed, this choice is motivated by the outstanding ability of transformer models to understand natural language, enabling them to accurately capture semantic meaning, context and interrelationships within textual data. Such

an ability proves crucial in accurately identifying instances of hate speech [20].

In addition, the versatility of transformer models enables them to be integrated with ensemble learning techniques, thus improving the method's performance and robustness [12]. Ensemble Learning, in particular, is used to alleviate the problems of variance and over-fitting associated with individual transformer models by leveraging various ensembling strategies [16].

To this end, we have fine-tuned seven state-of-the-art pre-trained language models (BERT, RoBERTa, Electra, DistilBERT, ALBERT, Large-BERT, XLMRoBERTa) on hate speech detection downstream task. We have applied various ensemble learning strategies (Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging) to combine the predictions of the seven individual models and improve the overall performance.

Extensive experiments have been conducted to assess the performance of the proposed method involving different parameters and strategies. Evaluation demonstrates that our method achieves promising results and outperforms competitive baselines in terms of various standard evaluation metrics.

The remainder of this paper is organized as follows. Section 2 discusses some related work. Section 3 presents the proposed hate speech detection method. Experiments and results are outlined and discussed in Section 4. The last section concludes the paper and provides insights for future work.

## 2 Related Work

Online hate speech is a pervasive and complex phenomenon that poses serious challenges for researchers, policymakers, and social media platforms. The widespread use of social media, especially during the COVID-19 pandemic, has contributed to the spread and escalation of hate speech across different languages, cultures and topics [2]. Consequently, researchers have shown significant interest in this problem [8]. Here, we briefly review some recent studies on hate speech and offensive language detection.

In the study proposed by Mozafari et al. [18], authors introduced a method for hate speech detection in social media posts. The proposed method leveraged BERT, a pre-trained language model, that has been fine-tuned with different strategies examining the effect of combining different layers, such as Bi-LSTM and CNN.

Experiments demonstrated that the proposed method overcame some of the challenges in the existing data. The best results were achieved by combining all pre-trained BERT layers with a CNN layer. Malik et al. [14] reviewed 14 classifiers based on shallow or Deep Learning, using different word representation methods.

They found that neural network-based classifiers combined with BERT, ELECTRA or ALBERT performed better than other methods. Their best models were BERT+CNN and ELECTRA+MLP, which achieved a weighted F1-score of 91% on Davidson dataset [5]. In the study proposed by Kovács et al. [9], authors proposed a model that combined RoBERTa and FastText with CNNs and RNNs and obtained weighted F1-score of 72%.

The closest recent work to our strategy is the one proposed by Magnossão et al. [12], in which authors tested six transformer models and two basic ensemble methods but only on Arabic language. Their best results were achieved by using the ensemble learning majority vote, and reported an F1-score of 0.60 and Accuracy of 0.86 on the test set. More recently, the method of Mazari et al. [15] used FastText and GloVe word embeddings with Bi-LSTM and Bi-GRU to build Deep Learning models.

They combined the obtained models with BERT to create ensemble learning architectures and reported a ROC-AUC score of 98.63% on social media data provided on Kaggle. Our proposal introduces a new strategy for hate speech detection that uses base transformers for classification and ensemble learning.

We examined the recent studies on this topic and found that most of them did not use only base transformers for classification. We found that most of the existing work integrated machine learning classifiers or transformers with other complex deep learning models.

**Table 1.** Used transformer models

| Transformer | Developer |
|---|---|
| BERT | Google Research |
| Electra | |
| ALBERT | |
| Large-BERT | |
| RoBERTa | Facebook AI Research |
| XLM-RoBERTa | |
| DistilBERT | Hugging Face |

## 3 Proposed Method

In this section, we describe our hate speech detection method, presenting the transformer models and ensemble learning strategies we applied. Figure 1 depicts the general architecture of our proposal. The proposed method can be divided into three main steps : (1) Data Pre-processing, (2) Fine-tuning Transformers, and (3) Ensemble Learning.

### 3.1 Data Pre-Processing

Data pre-processing is an essential and necessary phase before dealing with text data. its aim is to make raw data more suitable and understandable for machine learning or data analysis algorithms. This way, the model algorithm can work effectively and extract meaningful features and patterns from data [13].

In our method, this step essentially involves data cleaning, lowercase conversion and tokenization. Given our dataset, the result of this step consists of two distinct sets of data: a train set that forms the input of our primer models, and a separate test set for evaluation purposes.

### 3.2 Fine-tuning Transformers

Transformer models [22] have revolutionized the field of natural language processing (NLP). However, training these models from scratch requires a vast amount of data and substantial computational resources. Fortunately, there are various state-of-the-art pre-trained transformer models which are publicly available and can be fine-tuned for specific tasks. These models have different sizes: (i) Base, (ii) Medium, and (iii) Large, based on the number of parameters they can learn.

For our task, we chose, as shown in Table 1, seven of the publicly available transformer models, which have demonstrated significant results on various NLP tasks. The transformer models were imported from the open-source HuggingFace transformer library[1] and used for fine-tuning. Indeed, the fine-tuning process involves updating weights and adjusting the appropriate hyper-parameters for each model separately to optimize their performance.

The models were trained on the train set to learn task specific patterns and features. Once the training process is complete, we obtain a set of fine-tuned models, which are capable of classifying new tweets from the test set and producing level 1 predicted labels.

### 3.3 Ensemble Learning

Ensemble learning workflow is to train a set of individual base learners first and then combining them to improve results via some ensembling strategies [24]. In our method, we ensembled all of the fine-tuned transformer models from Table 1 using six different strategies of ensemble learning: Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging. We outline each technique employed in the following subsections.

#### 3.3.1 Majority Voting

Majority voting can be approached in different ways, depending on the level of agreement required among the base transformer models: unanimous voting, simple majority, max voting [17]. We chose max voting, which takes the prediction that has maximum votes of the set of classifiers, according to the following formula:

$$\sum_{t=1}^{T} d_{t,c*} = \max_{c} \sum_{t=1}^{T} d_{t,c}, \tag{1}$$

[1] huggingface.co/docs/transformers/

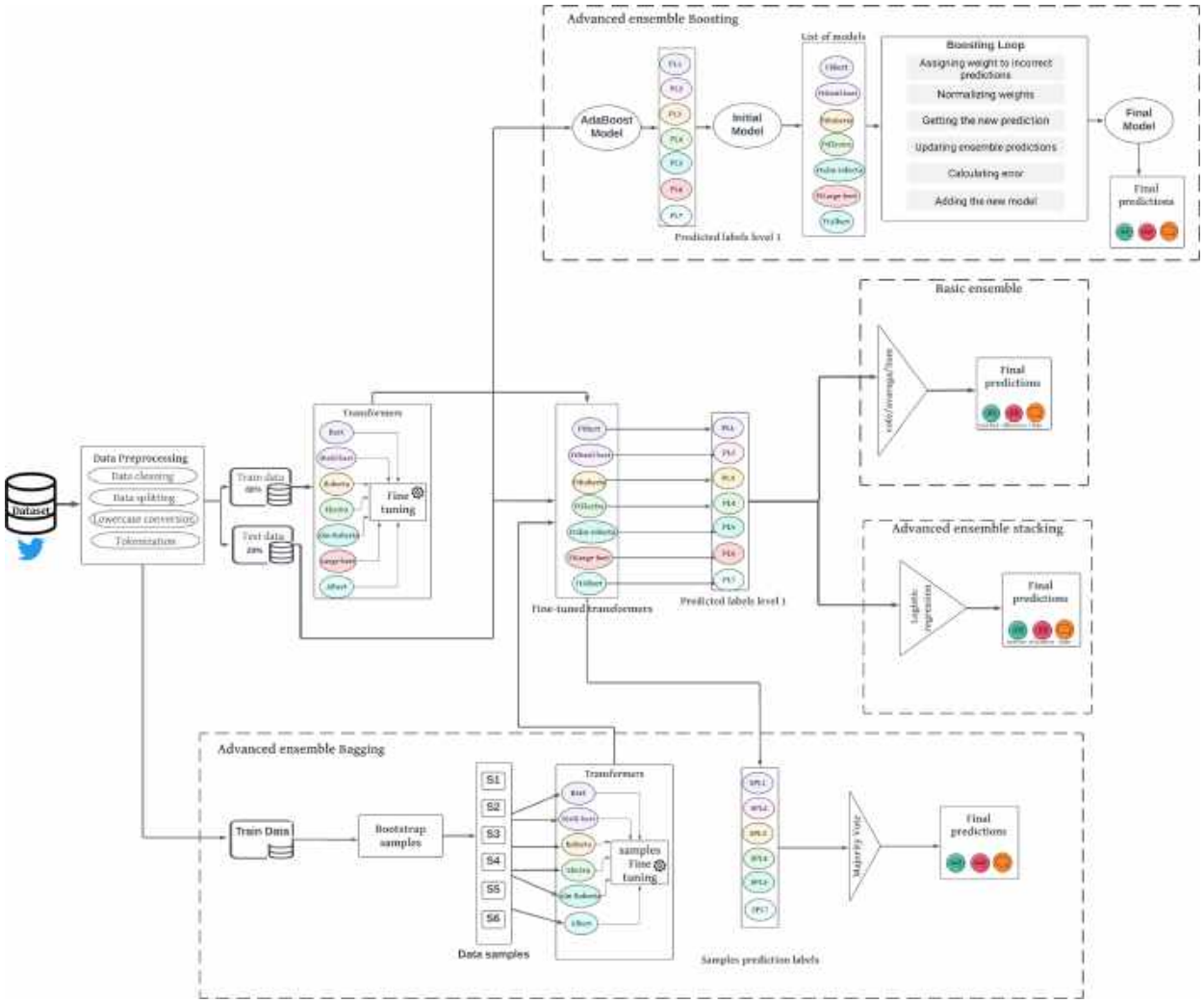



**Fig. 1.** Overall architecture of the proposed method

where $d_{t,c}$ represents the prediction of model $t$ for the class $c$. The resulting class will be determined based on the majority of votes from the models.

### 3.3.2 Averaging

Averaging takes the average value of all models' probabilities. It can be mathematically presented as follows:

$$P_i^j = \text{softmax}^j(O_i) = \frac{O_i^j}{\sum_{k=1}^{K} \exp(O_k^j)}. \qquad (2)$$

Using Softmax function, where $P_i^j$ represents the probability outcome of the $i^{th}$ unit on the $j^{th}$ base model, $O_i^j$ is the output of the $i^{th}$ unite of the $j^{th}$ base model and $K$ is the number of classes [7].

### 3.3.3 Highest-sum

In our highest-sum strategy, final predictions were made by selecting the classes with the highest sum of predicted probabilities or confidence scores from each one of the transformer models.

**Table 2.** Fine-tuning results for seven transformers models. The highest values of each model are marked in bold

| Transformers | Hyper-parameters | Accuracy | F1-score |
|---|---|---|---|
| BERT | Batch Size = 8 | 91.24% | 90.03% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 16 | 91.56% | 90.56% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91.22% | 90.14% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91.82% | 91,22% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 5 | | |
| RoBERTa | Batch Size = 8 | 91,40% | 90,70% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 16 | 90.67% | 90.14% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91,56% | 90.74% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 5 | | |
| DistilBERT | Batch Size = 16 | 91.93% | 90.76% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 92.17% | 91,53% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| XLM-RoBERTa | Batch Size = 16 | 90.23% | 90.16% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91,74% | 90.49% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| Electra | Batch Size = 16 | 90.98% | 88,62% |
| | Learning Rate = 5e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91,56% | 90,62% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| ALBERT | Batch Size = 16 | 90.21% | 89.39% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| | Batch Size = 32 | 91,14% | 90,55% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |
| Large-BERT | Batch Size = 16 | 91,95% | 91,00% |
| | Learning Rate = 2e-5 | | |
| | Epochs = 3 | | |

#### 3.3.4 Stacking

Performing stacking involves combining different fine-tuned base models to reduce their errors. The predictions from each model are stacked together and used as input to a final meta-model, which is a logistic regression in our case.

The meta-model was trained using cross-validation on the level 1 predictions of the base models and learned how to best combine them and produced the final predictions. Stacking can be represented as shown in formula 3, where $h_t$ represent our base models trained on data $D$ resulting the predictions $z$ that was fed to the meta-model:

$$z = \text{stack}\{h_t(D)\}. \tag{3}$$

#### 3.3.5 Boosting

Boosting is another ensemble learning technique that aims to improve the performance of a single ”weak” learner by successively training multiple models, where each subsequent model focuses on misclassified samples or samples with high residual errors compared to previous models [23].

#### 3.3.6 Bagging

Bagging, or Bootstrap Aggregating, is an ensemble learning technique that involves training multiple models independently on different subsets of the training data. After training all the base models, their predictions are then combined to make a final prediction.

## 4 Experiments and Results

In this section, we first describe our dataset and experimental settings. Then we discuss the obtained results.

### 4.1 Dataset

For dataset selection, we have carefully considered various factors such as the diversity of content, annotation quality, and the relevance of the dataset to real-world social media scenarios. We therefore chose to work with the Davidson dataset [5] for our experiments, which was specially created for the purpose of detecting hate speech and offensive language on social media.

It serves as a valuable resource in the field of harmful content detection, providing researchers with a diverse collection of annotated comments extracted from Twitter. The dataset is publicly available and it contains a significant number of tweets in English, ensuring a substantial amount of data for analysis. It consists of approximately 25,296 labeled instances classified into three classes: Hate speech (5.70%), Offensive Language (77.60%) or Neither (16.70%).

### 4.2 Experimental Setup

We present a comprehensive investigation into the application of fine-tuning techniques with transformer models to enhance the classification performance across the variety of models that we used. One of the benefits of transformers is that they do not require a lot of pre-processing in order to make the data understandable. For our method, we adopted the main following steps for pre-processing our dataset:

- Data Cleaning: includes removing useless columns, removing URLs, removing mentions and users, replacing numbers with $\langle \text{numbers} \rangle$, removing special characters, etc.
- Using Sklearn library to apply stratified splitting of the dataset into an 80:20 ratio, with 80% as Train set and 20% as Test set.
- Converting data into lower case format.
- Tokenizing data: each one of the transformer models has its own tokenizer from the pre-trained transformer library, which has its own characteristics and allows to do tokenization or reverse tokenization.

Afterword, the pre-processing step is followed by a fine-tuning process. For each model, we meticulously evaluated their Accuracy and F1-score on the Davidson dataset, by leveraging specific hyper-parameters, notably employing several batch sizes, as well as multiple learning rates. We employed a max length padding of 128, which is the nearest number to the maximum tokenized sequence length of 125. The dropout probability was set to 0.1. A range of 3 to 5 epochs was employed for training. Subsequently, to aggregate the outcomes of all fine-tuned models, we adopted some of the comprehensive ensemble learning techniques.

### 4.3 Results and Discussion

In Table 2, we present the results of a comparative study in which we assessed how various fine-tuning strategies impact the performance of our transformer models. We display the obtained results of the different applied parameters in terms of Accuracy and F1-score. The entire experimental process was conducted on Google Collaboratory, which provides free but limited daily usage of a Tesla 4 (T4) GPU and 12 GB RAM.

We began our experiments with BERT, one of the most well-known transformers for NLP tasks [6]. After examining multiple previous research and conducting personal experiments, we found that BERT achieves its best results when using a high number of batches. As shown in Table 2, we increased the batch-size from 8 to 32 and used a medium learning rate (2e-5), which yielded positive results.

We observed similar results for DistilBERT and ALBERT, which are smaller versions of the BERT model, designed to be more efficient, reduce model size and maintain similar performance with a lighter weight in terms of computational resources. As we can see, the best results were obtained with DistilBERT, reaching an accuracy of over 92%.

RoBERTa and XLM-RoBERTa models, developed by Facebook [1], are also variants of BERT which have been pre-trained longer over more data with bigger batches [10]. We chose to increase the batch size for these models as well, which had a positive impact on our results.

**Table 3.** Ensemble learning strategies test results compared to literature baselines

| | Accuracy | F1-Score | Precision |
|---|---|---|---|
| Davidson et al. | – | 90% | 91% |
| Waseem et al. | – | 89% | – |
| Mukherjee et Das | – | 90,84% | – |
| Majority vote | 92,35% | 91,47% | 91,40% |
| Highest-sum | 92,25% | 91,26% | 91,26% |
| Averaging | 92,15% | 91,10% | 91,22% |
| Boosting | 92,33% | 91,39% | 91,37% |
| Bagging | 91,80% | 91,26% | 91,02% |
| Stacking | **92,47%** | **91,80%** | **91,64%** |

ELECTRA, designed to be more sample-efficient and quicker than other pre-trained models [3], also performed well when we adjusted the learning rate and used two different batch-sizes.

Large-BERT was the only model that restricted our experiments to a maximum batch-size of 16 due to its large size and the limitations of our experimental environment. Despite these limitations, we still obtained promising results.

The obtained results demonstrate the effectiveness of fine-tuning transformer models for the hate speech detection task. As we can see, all the seven tested models performed well in terms of test accuracy and F1-score.

After obtaining the results of the transformers fine-tuning phase, we show in Table 3 how different ensemble learning strategies impact the performance of our final models.

In order to evaluate and compare the effectiveness of the proposed method, we consider the following works as reference baselines, which were chosen according to the Davidson dataset:

– **Davidson et al. [5]**: the method which resulted in the creation of the actual dataset.

– **Waseem et al. [21]**: the proposed method trained a machine learning model using a multi-task learning (MTL).

– **Mukherjee and Das [19]**: method in which authors used transformers and adopted RoBERTa model as their best result.

Results are reported on the test dataset in terms of three evaluation metrics: Accuracy, F1-score and Precision. As we can see, by applying ensemble learning techniques, we can observe an overall improvement in performance. We began by applying basic ensemble learning techniques, namely: Majority Voting, Highest-sum and Averaging.

From the results obtained, we note that the application of all three techniques led to a further improvement in performance compared to the results achieved by each model individually, with all three techniques exceeding 92% in terms of Accuracy. Furthermore, we can observe that our method outperforms the reference baselines in terms of all evaluation metrics.

In the next set of experiments, we turned to more advanced ensemble learning techniques, namely: Stacking, Boosting and Bagging. According to the results obtained, the Stacking method further enhanced performance, achieving an accuracy of 92.47% and an F1-score of 91.80%, in addition to a precision of 91.64%.

It is worth noting that applying the Bagging method required a large amount of memory, as all the models were trained simultaneously. Due to this memory constraint, it was not feasible to train all the models with this method.

Consequently, we had to exclude the ”Large-BERT” model due to its large size and keep the remaining six models. We believe that the results can be further improved with the Bagging method using all seven transformers.

## 5 Conclusion and Future Work

In this paper, we focused on the task of Offensive Language and Hate Speech detection using transformers and ensemble learning. We employed seven state-of-the-art pre-trained language models such as BERT and RoBERTa, as well as six ensembling strategies.

Experiments demonstrated the effectiveness of the proposed method in addressing the

challenges posed by hate content. The obtained results highlighted the performance of fine-tuned transformers and the significance of ensemble learning. These findings provide additional support for the efficacy of transformer-based techniques in NLP tasks, reinforcing their effectiveness and applicability. Results also showed that the use of ensembling strategies yielded better performance than using transformers alone.

This observation is exemplified by the successful application of different ensemble learning techniques, highlighting the power of combining multiple models to achieve better results. As future work, we plan to explore other transformer models and incorporate more diverse datasets as well as applying further hyper-parameters optimization to improve the detection performance. Another promising research direction is to study the use of the proposed method in other NLP tasks, such as Fake News Detection (FND).

## References

1. **Chernyavskiy, A., Ilvovsky, D., Nakov, P. (2021).** Transformers: "The end of history" for natural language processing? Springer International Publishing, pp. 677–693. DOI: 10.1007/978-3-030-86523-8_41.
2. **Cinelli, M., Pelicon, A., Mozetič, I., Quattrociocchi, W., Novak, P. K., Zollo, F. (2021).** Dynamics of online hate and misinformation. Scientific Reports, Vol. 11, No. 1. DOI: 10.1038/s41598-021-01487-w.
3. **Clark, K., Luong, M. T., Le, Q. V., Manning, C. D. (2020).** ELECTRA: Pre-training text encoders as discriminators rather than generators. Proceedings of the 8th International Conference on Learning Representations, pp. 1–18. DOI: 10.48550/ARXIV.2003.10555.
4. **Cramer, R. J., Fording, R. C., Gerstenfeld, P., Kehn, A., Marsden, J., Deitle, C., King, A., Smart, S., Nobles, M. R. (2020).** Hate-motivated behavior: impacts, risk factors, and interventions. Health Affairs, Health Policy Brief, Vol. 9, pp. 1–6. DOI: 10.1377/hpb20200929.601434.
5. **Davidson, T., Warmsley, D., Macy, M., Weber, I. (2017).** Automated hate speech detection and the problem of offensive language. Vol. 11, No. 1, pp. 512–515. DOI: 10.1609/icwsm.v11i1.14955.
6. **Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2019).** BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.
7. **Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., Suganthan, P. N. (2022).** Ensemble deep learning: A review. Engineering Applications of Artificial Intelligence, Vol. 115, pp. 105151. DOI: 10.1016/j.engappai.2022.105151.
8. **Jahan, M. S., Oussalah, M. (2023).** A systematic review of hate speech automatic detection using natural language processing. Neurocomputing, Vol. 546, pp. 126232. DOI: 10.1016/j.neucom.2023.126232.
9. **Kovács, G., Alonso, P., Saini, R. (2021).** Challenges of hate speech detection in social media: Data scarcity, and leveraging external resources. SN Computer Science, Vol. 2, No. 2. DOI: 10.1007/s42979-021-00457-3.
10. **Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V. (2019).** RoBERTa: A robustly optimized BERT pretraining approach. Proceedings of the International Conference on Learning Representations, pp. 1–15. DOI: 10.48550/arXiv.1907.11692.
11. **MacAvaney, S., Yao, H. R., Yang, E., Russell, K., Goharian, N., Frieder, O. (2019).** Hate speech detection: Challenges and solutions. PLOS ONE, Vol. 14, No. 8, pp. e0221152. DOI: 10.1371/journal.pone.0221152.
12. **Magnossão-de-Paula, A. F., Bensalem, I., Rosso, P., Zaghouani, W. (2023).**

Transformers and ensemble methods: A solution for hate speech detection in Arabic languages. Proceedings of the Centre for Electronic Ubiquitous Research Workshop, pp. 1–7. DOI: 10.48550/arXiv.2303.09823.

13. **Maharana, K., Mondal, S., Nemade, B. (2022).** A review: Data pre-processing and data augmentation techniques. Proceedings of the Global Transitions, Vol. 3, No. 1, pp. 91–99. DOI: 10.1016/j.gltp.2022.04.020.
14. **Malik, J. S., Pang, G., Hengel, A. V. D. (2022).** Deep learning for hate speech detection: A comparative study. arXiv. DOI: 10.48550/arXiv.2202.09517.
15. **Mazari, A. C., Boudoukhani, N., Djeffal, A. (2023).** Bert-based ensemble learning for multi-aspect hate speech detection. Cluster Computing, Vol. 27, No. 1, pp. 325–339. DOI: 10.1007/s10586-022-03956-x.
16. **Mnassri, K., Rajapaksha, P., Farahbakhsh, R., Crespi, N. (2022).** Bert-based ensemble approaches for hate speech detection. IEEE Global Communications Conference, pp. 4649–4654. DOI: 10.1109/GLOBECOM48099.2022.10001325.
17. **Mohammed, A., Kora, R. (2023).** A comprehensive review on ensemble deep learning: Opportunities and challenges. Journal of King Saud University - Computer and Information Sciences, Vol. 35, No. 2, pp. 757–774. DOI: 10.1016/j.jksuci.2023.01.014.
18. **Mozafari, M., Farahbakhsh, R., Crespi, N. (2019).** A bert-based transfer learning approach for hate speech detection in online social media. Complex Networks and Their Applications VIII, pp. 928–940. DOI: 10.1007/978-3-030-36687-2_77.
19. **Mukherjee, S., Das, S. (2021).** Application of transformer-based language models to detect hate speech in social media. Journal of Computational and Cognitive Engineering, Vol. 2, No. 4, pp. 278–286. DOI: 10.47852/bonviewjcce2022010102.
20. **Roy, S. G., Narayan, U., Raha, T., Abid, Z., Varma, V. (2021).** Leveraging multilingual transformers for hate speech detection. Working Notes of Forum for Information Retrieval Evaluation, pp. 128–138. DOI: 10.48550/ARXIV.2101.03207.
21. **Talat, Z., Thorne, J., Bingel, J. (2018).** Bridging the gaps: Multi task learning for domain transfer of hate speech detection: Multi-task learning for domain transfer of hate speech detection. Springer International Publishing, pp. 29–55. DOI: 10.1007/978-3-319-78583-7_3.
22. **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017).** Attention is all you need. Advances in Neural Information Processing Systems, Vol. 30, pp. 5998–6008.
23. **Zhang, C., Ma, Y. (2012).** Ensemble machine learning: Methods and applications. Springer. DOI: 10.1007/978-1-4419-9326-7.
24. **Zhou, Z. H. (2009).** Ensemble Learning. Springer US, pp. 270–273. DOI: 10.1007/978-0-387-73003-5_293.

*Article received on 17/10/2023; accepted on 06/05/2024.*
**Corresponding author is Billel Aklouche.*

ISSN 2007-9737

# Sorting Algorithms Comparison on FPGA and Intel i7 Architectures

Yomna Ben-Jmaa[1,*], David Duvivier[2]

[1] University of Sfax, ReDCAD Laboratory,
Tunisia

[2] Polytechnic University of Hauts-de-France, LAMIH Laboratory,
France

yomna.benjmaa@redcad.org, david.duvivier@uphf.fr

**Abstract.** Sorting is an essential operation in many real-time applications, and choosing the right architecture to perform sorting tasks can significantly impact performance. This study aims to provide a comprehensive literature review on the implementation of several sorting algorithms on Intel i7 and FPGA architectures. On these architectures, we analyze and compare the performance and temporal stability of five different sorting algorithms: quick-sort, heap-sort, shell-sort, merge-sort, and tim-sort. Their performance are evaluated in terms of average and standard deviation of computational times on different numbers of elements ranging from 8 to 4096. The maximum number of elements to be sorted is set to 4096, as this is the number provided for a real-time decision support system as solutions to be sorted. However, our study provides insights into the performance of different sorting algorithms on different architectures, which can be useful for selecting the appropriate architecture for real-time computing applications in decision support systems.

**Keywords.** Field programmable gate array (FPGA), computational times, sorting algorithm.

## 1 Introduction

Nowadays, Embedded electronic systems have become ubiquitous in various sectors of activity, from transportation to healthcare, from consumer electronics to industrial automation.

One of the primary objectives of designing embedded electronic systems is to ensure their reliability and cost-effectiveness while performing complex tasks that meet the constraints of time, energy consumption, and manufacturing cost. Focusing on the scope of this paper, it is possible to mention as an example that flight plan planning algorithms used to find the shortest route or propose real-time avoidance trajectories incorporate sorting algorithms.

Indeed, sorting algorithms [29, 43, 1, 2] (sorts for short) play a vital role in the design and performance of embedded systems. In these systems, sorts are designed to meet the constraints of time and energy consumption. The choice of a sort depends on the nature of the data, the constraints, and the desired performance.

For example, bubble-sort or insertion-sort algorithms are easy to implement but are not suitable for large datasets or real-time applications. Conversely, quick-sort [37, 19], heap-sort [39], tim-sort [6, 22] and merge-sort [31, 19] algorithms are more complex and require more processing power but can handle large datasets and real-time applications. As technology continues to evolve, the demand for efficient and reliable embedded systems will only increase, and sorts will continue to play a vital role in meeting these demands.

Among these systems, FPGAs (Field-Programmable Gate Array) have emerged as an interesting alternative to accelerate software applications. FPGAs are integrated circuits that can be programmed to perform specific functions. They offer high performance and energy efficiency,

making them suitable for real-time computing applications. Considering sorting operations, one approach is to use the FPGA's built-in hardware resources to implement the sorts directly.

These algorithms can be optimized for the FPGA architecture, resulting in faster and more efficient sorting. Another approach is to use the FPGA to offload sorting operations from the CPU. In this approach, the CPU sends the data to be sorted to the FPGA, which sorts and returns the sorted data to the CPU.

This approach can significantly reduce the processing time required for sorting, as the FPGA can handle large data sets in parallel. One challenge in using FPGAs is the programming complexity since FPGA programming requires specialized knowledge in HDLs (Hardware Description Languages) such as Verilog and VHDL (very-high-speed integrated circuit hardware description language).

However, available tools and libraries can ease the programming process, making it more accessible to software developers. Moreover, CPUs can certainly be used in embedded systems. In many cases, embedded systems require a combination of processing power and low power consumption, which can be challenging to achieve with CPUs alone. To meet these requirements, CPUs are often used in conjunction with other components, such as FPGA's microcontrollers, memories, communication interfaces, sensors, and other peripherals.

These components are integrated into a complete embedded system, which is designed to meet the specific needs of the application. The choice between CPUs and FPGAs really depends on the specific needs of the application. CPUs are great for applications that require flexibility and versatility, as they can be programmed to handle a wide range of tasks. On the other hand, FPGAs are designed for specific tasks, and are optimized for performance in those tasks.

This makes them ideal for applications that require high performance. While they may be more expensive to produce and harder to program than CPUs, their performance benefits can make them the best choice for some applications.

The objective of this paper is to compare optimized hardware and software implementations of heap-sort, shell-sort, quick-sort, tim-sort and merge-sort on FPGA and Intel i7 architectures using a limited number of elements ranging from 8 to 4096. Indeed, contrary to most of related works in the literature, there are 4096 elements at most because they are provided to a real-time decision support system as solutions to be sorted.

This is highlighted by the work of K. Nikolajevic [35], whose thesis aims to tackle the challenging problem of reducing operational accidents in avionics systems. As part of the collision avoidance alarm system, sorting these solutions efficiently is crucial for real-time decision making in avionics systems to select the best actions to avoid accidents.

To be more precise, each solution (i.e. a short-term path to follow in an avionics application) is identified by a unique 32-bit integer, so-called index, and evaluated on the basis of various performance criteria (such as distance...). Consequently, index-sorts are used in the real-life application. However, sorted elements constitute permutations of integers in this paper to simplify the problem. In general terms, the main contributions of this paper are as follows:

– Analysis and comparisons of the performance of five sorts: quick-sort, heap-sort, shell-sort, merge-sort, and tim-sort. Their performances are evaluated in terms of average and standard deviation of computational times on Intel i7 and FPGA architectures. These measurements are refined by statistical tests.
– Temporal stability analysis of the sorts: In addition to ranking the performance of the sorts on each platform, a statistical analysis is carried out on the basis of "boxplot-like" statistical measurements to assess the temporal stability of these algorithms.

The paper is structured as follows: Section 2 presents a state of the art on various sorts and several applications using different platforms (CPU, FPGA). Section 3 shows our experimental results. Section 4 gathers our conclusions and some of our future works.

## 2 Related Works

Over the past few decades, there has been a significant amount of research conducted on sorts. While many of these studies have focused on accelerating sorts in heterogeneous computing systems, there has also been a focus on reducing computational time, power consumption, and hardware resources. This literature review section provides an overview of numerous studies on both FPGA-based and CPU-based sorts that aim to improve acceleration performance.

### 2.1 Hardware Acceleration Methods for Embedded Systems (FPGA)

FPGAs are greatly flexible and customizable to meet the specific requirements of different applications. This flexibility allows FPGAs to be optimized either to perform high-performance parallel processing and data streaming in applications that require high throughput, or for fast response times in applications that require low latency, or finally for energy efficiency in applications that require low-power consumption.

The traditional development of FPGA-based applications has been mainly based on highly specialized register transfer level (RTL) designs [44, 18, 42, 5, 13, 34, 20]. High-Level Synthesis (HLS) allows increasing design productivity and detaching the algorithm from architecture [16]. In our case, the vivado HLS tool is used to generate hardware accelerators from C language.

Ben Jmaa et al. [9] proposed an efficient hardware implementation for different sorts using high-level descriptions in a zynq-7000 platform. The authors compared the performance of the algorithms in terms of computational time, standard deviation and resource utilization. The results showed that the selection-sort was 1.01-1.23 times faster than other algorithms for less than 64 elements; otherwise, tim-sort was the best algorithm. Kobayashi et al. [26, 27] detailed a new approach to reducing FPGA programming costs while maintaining high levels of performance.

Their sorting library can use OpenCL for FPGA. This approach consumed at least twice the hardware resources of the merge-sort method restructured for the OpenCL programming model for FPGA. However, it operated at a frequency 1.08 times higher and had a sorting throughput three orders of magnitude greater than the baseline.

Chen et al. [14] proposed a sample-sort algorithm on a server with a PCIe-connected FPGA to sort large data sets. The prototype system was implemented using Verilog HDL on Amazon Web Services (AWS) FPGA instances equipped with Xilinx Virtex UltraScale+ FPGAs.

The authors demonstrated that this system can sort 230 key-value records 37.4 times faster than GNU parallel sort running on a CPU with 8 threads. However, their method assumed collaboration between the CPU and FPGA, and the sorting performance was ultimately limited by the PCIe bandwidth, which was 7.2 GB/s as reported in [14]. Moreover, their method was not suitable for implementing FPGA-centric applications that require sorting due to its structure.

Shinyamada et al. [38] evaluated the impact of various sorts on high-level synthesized image processing hardware. The results showed that bubble-sort and odd-even-merge-sort were the fastest algorithms, as they were able to achieve pipeline processing. Conversely, selection-sort was not able to achieve ideal pipeline processing, and its performance was not as good as the former two algorithms. Concludingly, optimizing sorts can have a significant impact on overall image processing performance.

Moghaddamfar et al. [32] conducted a comparative analysis of OpenCL and RTL-based implementations of a heap-sort that merges sorted runs. The results showed that while both implementations required comparable development effort, their RTL implementations of critical primitives achieved four times better performance and used only half as much FPGA resources compared to the OpenCL implementation.

This highlighted the importance of carefully selecting the programming language and implementation approach when developing algorithms for FPGA-based systems. The study suggested that RTL-based implementations can provide significant performance benefits and

resource efficiency compared to higher-level language implementations like OpenCL.

Chen et al. [15] proposed a novel hybrid pipelined sorting architecture based on a bitonic-sorter and several cascaded sorting units. This sorting architecture achieved a balanced trade-off between resource utilization and throughput, as well as between throughput and power consumption.

Specifically, the architecture was both resource and energy efficient in terms of the throughput-to-resource ratio and the throughput-to-power ratio. This study highlighted the importance of designing sorting architectures that maximize data parallelism to achieve increased throughput and reduced latency.

Montesdeoca et al. [33] monitored by a network of 40 $CO_2$ sensors and performed real-time sorting of all the data via bubble-sort and insertion-sort on FPGA. The results showed that insertion-sort was faster than bubble-sort, but it consumed more hardware resources in the FPGA, illustrating the importance of the trade-off between speed and resource utilization when selecting a sort for real-time applications.

In [3], the authors evaluated multithreaded sorts on a 32-core reconfigurable architecture with embedded real-time Linux support. The architecture consisted of NIOS II/f soft cores and was implemented on an FPGA. The authors proposed a new approach for performance evaluation of a soft multithreaded multicore architecture conducted in real-time.

This approach was based on the recursive generation and execution of sorts such as merge-sort and quick-sort. The architecture was capable of achieving high parallelism and throughput while maintaining low latency. The architecture outperformed the others in terms of speed and scalability. In [1], the authors focused on hardware implementing bubble-sort, selection-sort, insertion-sort, merge-sort, bitonic-sort and odd-even-merge-sort using FPGA in synchronous and pipelined architectures. The authors compared these implementations in terms of computational time and area.

They showed that non-pipelined bitonic-sort and non-pipelined odd-even-merge-sort had the best performance in terms of computational time, while the non-pipelined selection-sort and non-pipelined insertion-sort had the lowest area of synchronous architecture.

For pipelined architectures, bitonic-sort and odd-even-merge-sort had much lower computational time when implemented in hardware. Additionally, odd-even-merge-sort was found to be the smallest in terms of area. While bitonic-merge-sort was slightly larger in area and slower in execution than odd-even-merge-sort.

Lobo et al. [31] compared five merge-sorts (serial-merge-sort, parallel-merge-sort, bitonic-merge-sort, odd-even-merge-sort and the modified-merge-sort) in terms of resource utilization, delay and area on FPGA. The results showed that the serial and parallel merge use the highest amount of resource utilization compared to bitonic-merge, odd-even-merge and modified-merge.

Also, the parallel-merge algorithm was much faster than a serial-merge algorithm. In addition, the odd-even and modified-merge had a very close value of the area used while bitonic-merge had as slightly higher value.

Abdelrasoul et al. [2] proposed an index and sort algorithm (IaSA) based on an FPGA (vertex-5 series) in a pipelined sequential structure using Verilog HDL. The results showed that, for various data set sizes, IaSA performed best in terms of computational time.

In [24], the authors presented a column-sort, mapped on an HBM (High Bandwidth Memory)-enabled FPGA. The approach utilized computational pipelines, hardware-efficient interconnection networks and several optimizations to achieve high-throughput sorting. The results showed that their optimized design yields 14.8×, 4.73× and 2.18× speedup compared with state-of-the-art implementations on CPU.

– Lines 5 to 8: First step of the algorithm, check for external variations from the computational environment.

– Line 8: badRSDR$[s] \leftarrow$ true if unstable computational environment (let's try to continue anyway).

**Table 1.** Review of sorting algorithms on different platforms

| Approach | Platforms | | | Algorithms | | | HLS | Optimization | Application domain |
|---|---|---|---|---|---|---|---|---|---|
| Ref | FPGA | Intel CPU | ARM | Sort-name | Complexity | Perf. criteria | | | |
| [26, 27] | Yes | No | No | Library sorting | O(n log(n)) | Resources utilization | Yes | yes (openCL) | Sorting data |
| [9] | Yes | No | No | bubble, selection | O(n$\hat{2}$) | Resources utilization, computational times, standard deviation | Yes | Yes | ITS |
| | | | | insertion, quick | O(n^2) | | | | |
| | | | | shell | O(n^(3/2)) | | | | |
| | | | | merge,heap,tim | O(n log(n) | | | | |
| [14] | Yes | No | No | sample | O(n log(n)) | Computational times | No | yes | Amazon web server |
| [38] | Yes | No | No | bubble, selection, | O(n^2) | Computational times | Yes | Yes | Image processing |
| | | | | odd-even-merge | | | | | |
| [32] | Yes | No | No | heap, merge | O(n log(n)) | FPGA resource usage, development effort | No | Yes (OpenCL) | Sorting data |
| [15] | Yes | No | No | bitonic | O(log(n)^2) | FPGA resource usage, energy efficient | No | Yes (Pipeline) | Real World Application |
| [33] | Yes | No | No | bubble, insertion | O(n^2) | FPGA resource usage, computational times | No | Yes | Wireless sensor network on IoT |
| [3] | Yes | No | No | quick | O(n$\hat{2}$) | Parallelization efficiency, computational times | RTL | Yes (POSIX thread) | - |
| | | | | merge | O(n log(n)) | | | | |
| [1] | Yes | Yes | No | bubble, selection | O(n$\hat{2}$) | Computational times, area | No | Yes | Data processing |
| | | | | insertion | O(n^2) | | | | |
| | | | | merge | O(n log(n)) | | | | |
| [31] | Yes | No | No | serial merge, | O(n log(n)) | Resource utilization, delay, area | No | No | Particular application |
| | | | | parallel merge, | O(log(n)) | | | | |
| | | | | bitonic-merge, | O(log(n)^2) | | | | |
| | | | | odd-even-merge | O(log(n)^2) | | | | |
| | | | | modified merge | O(log(n)^2) | | | | |
| [2] | Yes | No | No | index | | Computational times | No | Yes (pipeline) | - |
| [24] | Yes | No | No | hyper, column | O(n log(n)) | External memory (HBM) | - | Yes | - |
| [11, 10] | Yes | No | Yes | insertion, quick | O(n$\hat{2}$) | Computational times, energy consumption, temporal stability | Yes | Yes | Intelligent systems (ITS) |
| | | | | shell | O(n^(3/2)) | | | | |
| | | | | heap, merge, tim | O(n log(n)) | | | | |
| [7] | Yes | No | Yes | heap | O(n log(n)) | Power, Speedup | No | Yes | Wavelet Based Image Coder |
| [8] | Yes | No | Yes | heap | O(n log(n)) | Power, Speedup | No | Yes | Image coding |
| [25] | Yes | No | Yes | heap | O(n log(n)) | Energy consumption | No | Yes | Embedded System |
| [12] | No | No | Yes | network sorting | O(log(n)^2) | Speedup | No | Yes | Commercial microchips |
| [30] | No | Yes | No | counting, | O(n) | Computational times | No | Yes (parallelism) | Sorting data |
| | | | | bucket, | O(n^2) | | | | |
| | | | | merge, | O(n log(n)) | | | | |
| [21] | No | Yes | No | bitonic, | O(log(n)$\hat{2}$) | Computational times, memory | No | Yes (OpenMp) | Sorting data |
| | | | | merge, | O( n log(n)) | | | | |
| [4] | No | Yes | No | insertion, quick | O(n$\hat{2}$) | Time, stability, memory space | No | No | Data base, Network, AI |
| | | | | bubble,selection | O(n^2) | | | | |
| | | | | merge | O(n log(n)) | | | | |
| [28] | No | Yes | No | insertion, stl | O(n^2) | Time | No | Yes | Computer science |
| Our work | Yes | Yes | No | quick | O(n$\hat{2}$) | Computational times, temporal stability | Yes | Yes (HLS directives) | Intelligent systems (ITS) |
| | | | | shell | O(n^(3/2)) | | | | |
| | | | | heap, merge, tim | O(n log(n)) | | | | |

**Algorithm 1:** Main steps of our DOE

**Input :** A given architecture (i7 or FPGA).
A list of sorts $\mathcal{S}$.
A value of $n \in \mathcal{N}$.
A set $\mathcal{P}_n$ of $P$ permutations $\pi_n$.

**Output:** A partial or total ranking of the sorts in the ordered list $\mathcal{S}'$.
Temporal stability of sorts in stable[·].
Out of range $\text{RSD}_R(s, \pi_n)$ in badRSDR[·].

1 $\mathcal{S}' \leftarrow \emptyset$
2 **foreach** sort $s \in \mathcal{S}$ **do**
3  stable$[s] \leftarrow$ false
4  badRSDR$[s] \leftarrow$ false
5  **foreach** permutation $\pi_n \in \mathcal{P}_n$ **do**
6  compute $\text{RSD}_R(s, \pi_n)$
7  **if** $RSD_R(s, \pi_n) > 5\%$ **then**
8  badRSDR$[s] \leftarrow$ true
9  compute $\mu_P(s)$, $\sigma_P(s)$, $\text{RSD}_P(s)$ and $\text{CI}(s)$
10  **if** $RSD_P(s) \leq 5\%$ **and** $Q2(s) \approx \mu_P(s)$ **then**
11  stable$[s] \leftarrow$ true
12  $\mathcal{S}' \leftarrow \mathcal{S}' + s$ ▷ `rank` $s$ `according to its` $\mu_P$
13  **else**
14  compute $IQR(s)$
15  **if** $s$ intersects with other sorts $\in \mathcal{S}'$ **then**
16  perform statistical tests
17  $\mathcal{S}' \leftarrow \mathcal{S}' + s$ ▷ `rank` $s$ `via p-values`
18  compute #$^+$outliers$(s)$ and %$^+$outliers$(s)$
19  adjust stable$[s]$ according to %$^+$ outliers$(s)$ if needed
20  adjust stable$[s]$ according to badRSDR$[s]$ if needed

– Lines 10 to 12: $\text{RSD}_P(s)$ is sufficiently low to avoid using sophisticated statistical tests to rank the sorts.

– Line 10: $\text{RSD}_P(s) \leq 5\%$ and $Q2(s) \approx \mu_P(s)$ suggest that the sort $s$ is stable and can be ranked via line 12.

– Line 12: $\text{RSD}_P(s) \leq 5\%$ is considered as sufficient for $\mu_P(s)$ to be statistically representative.

$\text{CI}(s)$ is compared with those of previously ranked sorts and used to assess whether a partial or a total ranking of the sorts is possible when inserting and ranking $s$ in $\mathcal{S}'$.

– Lines 13 to 17: In this step, statistical tests are required due to a high value of $\text{RSD}_P(s)$ or a significant difference between $Q2(s)$ and $\mu_P(s)$.

– Line 15: Current confidence interval $\text{CI}(s)$ (resp. $IQR(s)$) is compared with those of previously ranked sorts in $\mathcal{S}'$.

– Line 16: Statistical tests are used to rank the sorts if possible, temporal stability tests of $s$ are performed elsewhere.

– Line 17: At this point it is not always possible to strictly rank $s$. In this case, $\mathcal{S}'$ contains a partial ranking of the sorts.

– Lines 18 to 20: Check if sort $s$ is compatible with worst-case computational times (i.e. upper outliers). With line 10, this last step assesses/adjusts the stability of $s$ in line 19.

– Line 19: Set stable$[s]$ to false if %$^+$outliers$(s) > 5\%$.

– Line 20: Set stable$[s]$ to false if badRSDR$[s] =$ true to take unstable computational environment into account.

### 2.2 Software Acceleration Methods for Embedded Systems (ARM)

ARM processors offer advantages in terms of flexibility and ease of reconfigurable integration technology for a wide range of applications, from embedded systems to high-performance computing.

Compared to classical processors, soft-core processors like ARM allow for greater customization and adaptability because they can be easily programmed and reconfigured to meet the specific needs of a given application.

Additionally, soft-core processors can be integrated into a variety of reconfigurable technologies, such as FPGAs, allowing for even greater flexibility and performance optimization.

In [11, 10], the authors proposed a software implementation for insertion-sort, quick-sort, heap-sort, shell-sort, merge-sort, and tim-sort on an ARM Cortex A9. They compared the performance of these algorithms in terms of average and standard deviation of computational times, energy consumption, and temporal stability.

**Table 2.** Average and standard deviation of computational times on i7

| size/ns | tim-sort | merge-sort | heap-sort | shell-sort | quick-sort |
|---|---|---|---|---|---|
| 8 | 28.3 (2.2) | 83.7 (2.2) | 53.0 (3.5) | 38.2 (1.2) | 47.2 (2.7) |
| 16 | 49.6 (7.7) | 164.5 (4.8) | 121.3 (5.7) | 73.6 (2.9) | 87.0 (13.6) |
| 32 | 127.2 (38.0) | 346.1 (14.8) | 307.0 (13.8) | 168.1 (9.6) | 198.0 (72.6) |
| 64 | 565.4 (167.6) | 752.4 (45.8) | 682.2 (35.8) | 406.7 (24.6) | 507.8 (364.8) |
| 128 | 1285.8 (405.9) | 1685.4 (147.6) | 1582.5 (76.3) | 1031.2 (95.9) | 1374.3 (1399.5) |
| 256 | 3226.4 (1101.5) | 3622.4 (316.8) | 3954.4 (136.7) | 3716.8 (732.3) | 4003.8 (5396.8) |
| 512 | 9222.3 (3192.9) | 9374.1 (1355.0) | 9833.0 (318.2) | 15199.5 (4159.4) | 14837.7 (20382.2) |
| 1024 | 32528.8 (11768.7) | 33140.8 (7954.5) | 23720.2 (852.1) | 40367.4 (11791.8) | 53518.4 (80339.7) |
| 2048 | 84812.9 (31077.6) | 82910.9 (21840.3) | 56650.7 (2792.7) | 100684.9 (30593.3) | 182912.0 (329092.6) |
| 4096 | 189435.6 (69730.3) | 183894.0 (49724.5) | 129825.6 (7041.9) | 235353.3 (72545.8) | 641826.5 (1346751.2) |

The results demonstrated that shell-sort was the best algorithm, being 42.1% faster and even reaching up to 72% faster when the number of elements to be sorted is greater than 64.

However, when the number of elements is smaller than 64, tim-sort was the best algorithm. Additionally, shell-sort was the best algorithm in terms of standard deviation of computational times and energy consumption.

In [7], the authors proposed a hardware heap-sort implementation using FPGA of a wavelet based image coder. Their architecture provided up to 20.9% power reduction on the memories compared to the baseline implementation. Moreover, their architecture provided 13x speedup compared to ARM Cortex A9.

In [8], the authors introduced an adaptive heap-sort that was designed for an image coding implementation on FPGA with high throughput and scalable sorting. The authors compared its performance to an embedded ARM Cortex A9 running at 666 MHz.

Their architecture, running at 100 MHz, provided around 13 times the speedup while consuming 242 mW of average core dynamic power. In [12], the authors adapted a hybrid sort based on quick-sort and bitonic-sort. They employed bitonic-sort to handle small partitions/arrays with a vectorized partitioning implementation to divide these partitions.

Their approach required only an array of size O(log n) for recursive calls in the partitioning phase. They evaluated the performance on an ARM v8.2 (A64FX) and assessed their implementation by sorting/partitioning integers, double floating-point numbers, and key/value pairs of integers. The results showed an average speedup factor of four compared to the GNU C++ sort algorithm.

In [25], the performance and energy efficiency of hardware and software implementations of the heap-sort are compared. The results showed that the hardware implementation (Digilent Basys 3 Artix-7 FPGA) was more energy efficient, but slower than software implementation (ARM Cortex A72) due to a low clock frequency.

### 2.3 Software Acceleration Methods for Non Embedded CPU

In “standard” workstations, Central Processing Units (CPUs) can drastically increase the number of instructions processed per second, allowing the computer to perform more complex tasks or run more programs simultaneously. Indeed, recent CPUs have a potential for higher performance thanks to higher clock speeds, more cores, and improved SIMD instructions. However, these processors consume much more power than their embedded counterparts.

In [30], the authors presented optimized serial and parallel counting-sorts. They compared this

**Table 3.** $\text{RSD}_P(s)$ of sorting algorithms on i7

| size/% | tim-sort | merge-sort | heap-sort | shell-sort | quick-sort |
|---|---|---|---|---|---|
| 8 | 7.8 | 2.6 | 6.7 | 3.2 | 5.7 |
| 16 | 15.6 | 2.9 | 4.7 | 4.0 | 15.7 |
| 32 | 29.9 | 4.3 | 4.5 | 5.7 | 36.7 |
| 64 | 29.6 | 6.1 | 5.2 | 6.1 | 71.8 |
| 128 | 31.6 | 8.8 | 4.8 | 9.3 | 101.8 |
| 256 | 34.1 | 8.7 | 3.5 | 19.7 | 134.8 |
| 512 | 34.6 | 14.5 | 3.2 | 27.4 | 137.4 |
| 1024 | 36.2 | 24.0 | 3.6 | 29.2 | 150.1 |
| 2048 | 36.6 | 26.3 | 4.9 | 30.4 | 179.9 |
| 4096 | 36.8 | 27.0 | 5.4 | 30.8 | 209.8 |

sort to others such as bucket-sort and merge-sort, implementing both counting-sort and merge-sort on CPU and GPU. The results showed that the optimized counting-sort took only 6 ms to sort 100 million integers, being 23 times faster than the previous version.

In [21], the authors provided an analysis of current host-GPU data transfer mechanisms and explored methods for mitigating performance bottlenecks. They developed a heterogeneous CPU/GPU sort and demonstrated that while out-of-place GPU sorting achieved the best performance, an in-place sort further reduced some host-side bottlenecks.

In [4], the authors compared the grouping-comparison-sort (GCS) to selection-sort, quick-sort, insertion-sort, merge-sort and bubble-sort, using random input sequences. On an Intel Core 2 Duo E8400 @ 3.00 GHz (2 CPUs), the result revealed that for small input sizes, the performance of all six algorithms was almost comparable.

However, for larger input, quick-sort proved to be the fastest, while selection-sort was the slowest. GCS ranked as the third fastest for small input size (10000 elements) and the fifth fastest for large input size (30000 elements). In [28], the authors introduced a distribution sorting method that utilized a trained model of the empirical Cumulative Distribution Function of the data.

Additionally, they applied a deterministic sort that performed well on almost sorted arrays, such as insertion-sort. The performance was measured on an Intel Xeon Gold 6150 @ 2.70 GHz using up to one billion double-precision keys following a normal distribution.

Their approach achieved an average performance improvement of 3.38 times compared to the C++ STL-sort, which is an optimized hybrid of quick-sort, a 1.49 times improvement over sequential radix-sort, and a 5.54 times improvement over a C++ implementation of tim-sort, which is the default sorting function for Java and Python.

### 2.4 Synthesis of Related Work

Table 1 summarizes a literature review focusing on the use of several sorts on different platforms (FPGA, Intel CPU, ARM). This summary shows that the authors generally use sorts on FPGA to improve performance in terms of computational time and resource usage. Moreover, a majority of authors do not use HLS except for [26, 27, 38, 9, 11, 10]. In addition, the number of elements to sort is usually much larger than thousands of items. Contrary to these studies, our work is based on different implementations of sorts using HLS and compare the sorts on FPGA and Intel i7 in

**Table 4.** Ranking of sorts using $\mu_P(s)$ and CI$(s)$ on i7

| **size** | $\mathcal{S}'$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 8 | tim | $\prec$ | shell | $\prec$ | quick | $\prec$ | heap | $\prec$ merge |
| 16 | tim | $\prec$ | shell | $\prec$ | quick | $\prec$ | heap | $\prec$ merge |
| 32 | tim | $\prec$ | shell | $\prec$ | quick | $\prec$ | heap | $\prec$ merge |
| 64 | shell | ? | quick | ? | tim | $\prec$ | heap | $\prec$ merge |
| 128 | shell | $\prec$ | tim | ? | quick | ? | heap | $\prec$ merge |
| 256 | tim | ? | merge | ? | shell | ? | heap | ? quick |
| 512 | tim | ? | merge | ? | heap | ? | quick | ? shell |
| 1024 | heap | $\prec$ | tim | ? | merge | $\prec$ | shell | ? quick |
| 2048 | heap | $\prec$ | merge | ? | tim | ? | shell | ? quick |
| 4096 | heap | $\prec$ | merge | ? | tim | $\prec$ | shell | $\prec$ quick |

terms of computational time, resources utilization and temporal stability.

It is worth noting that while many authors are interested in the first two criteria, the number of papers concerning the stability of algorithms is much lower. Moreover, since an avionics application is targeted, deterministic sorting is required and the use recursive functions or dynamic memory allocations are forbidden contrary to many applications in the literature.

Similarly, parallel versions of sorts are not allowed because it is not possible to certify such algorithms on standard multicore architectures in our target avionics application.

A finding from this synthesis is that beyond its inherent complexity, the "best sort" depends on the number of elements to be sorted, the target architecture, the parallelization mode and the considered key performance indicators. In the following sections, we compare software and hardware implementations of sorts on Intel i7 and FPGA.

## 3 Experimental Results

In this study, the performances of five sorts $s \in \mathcal{S}$ are compared on Intel i7 and FPGA architectures: $\mathcal{S} = \{$heap-sort, quick-sort, merge-sort, shell-sort, tim-sort$\}$. The number of sorts corresponds to the cardinality of $\mathcal{S}$ and is denoted by $\Sigma = |\mathcal{S}|$. The sorts are evaluated in terms of computational times and temporal stability.

Considering sorting algorithms, the usual informal definition of the stability is the following: A sort is ideally stable if it maintains the relative order of elements with equal values. This means that whenever there are two elements $a$ and $b$ with the same value, the relative order of $a$ and $b$ is preserved by the sort.

However, this study focuses on temporal stability, defined as follows: A sort is temporally stable if its computational time is independent of the order of the elements to be sorted.

Although target application requires deterministic sorts, there is no guaranty (and no need) of usual stability in our implementations of the sorts since the resulting relative order of $a$ and $b$ (with the same value) will deterministically be the same after the sort but this is not necessarily the initial relative order of $a$ and $b$ (before the sort). Now that we have defined the evaluation criteria, next section describes our design of experiments.

### 3.1 Design of Experiments

In order to evaluate the performances of sorts, the average and standard-deviation of their computational times are studied using $n = 8$ to 4096 elements $n \in \mathcal{N} = \{$8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096$\}$, via $P =$47 permutations of $n$ integers generated using Lehmer's method [17]. For each value of $n$, this set of permutations $\mathcal{P}_n$ is used to characterize the temporal variations due to a sort $s$ in itself, i.e. its temporal stability.

For each permutation $\pi_n$ of size $n$, $R = 1000$ replications are used to identify the external variations coming from the "computational environment" (e.g., transmission error, operating system noise, I/O buffering). To further evaluate the stability of sorts, the Relative Standard Deviation (RSD) is calculated by dividing the standard deviation $\sigma$ by the average of computational times $\mu$ and expressed as a percentage:

$$\text{Relative\_Standard\_Deviation} = 100 \times \frac{\sigma}{\mu}. \quad (1)$$

**Table 5.** Ranking of sorts using statistical tests on i7

| **size** | $\mathcal{S}'$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 8 | tim | $\prec$ | shell | $\prec$ | quick | $\prec$ | heap | $\prec$ | merge |
| 16 | tim | $\prec$ | shell | $\prec$ | quick | $\prec$ | heap | $\prec$ | merge |
| 32 | tim | $\prec$ | shell | $\preceq$ | quick | $\prec$ | heap | $\prec$ | merge |
| 64 | quick | $\prec$ | shell | $\prec$ | tim | $\prec$ | heap | $\prec$ | merge |
| 128 | quick | $\prec$ | shell | $\prec$ | tim | $\prec$ | heap | $\prec$ | merge |
| 256 | quick | $\prec$ | tim | $\prec$ | merge | $\prec$ | shell | $\preceq$ | heap |
| 512 | quick | $\prec$ | merge | $\preceq$ | heap | $\prec$ | tim | $\prec$ | shell |
| 1024 | heap | $\preceq$ | quick | $\prec$ | merge | $\prec$ | tim | $\prec$ | shell |
| 2048 | heap | $\prec$ | quick | $\prec$ | merge | $\prec$ | tim | $\prec$ | shell |
| 4096 | heap | $\prec$ | quick | $\prec$ | merge | $\prec$ | tim | $\prec$ | shell |

It is a relative measure of the dispersion of data around the average. Ultimately, this ratio is used to compare the degree of variation from one sample to another, even if the means are different. For a sufficiently large number of items in a population (empirically $\geq$ 30), an RSD of less than 1% is considered "excellent" to make the average representative.

From a practical point of view, an RSD of 5% is generally considered acceptable. However, if RSD is greater than 5%, it is advisable to use statistical tests, possibly supplemented by graphical representations such as boxplots. Our study relies on two measures of RSD:

For each permutation $\pi_n$ of size $n$, $\text{RSD}_R(s, \pi_n)$ is computed over the set of $R$ replications of the same computational time measurement (i.e. for a given sort $s$, sorting the same permutation $\pi_n$).

Then, for each sort $s$, $\text{RSD}_P(s)$ is computed over the set $\mathcal{P}_n$ of $P$ permutations, considering – for each permutation $\pi_n$ – the average computational time $\mu_R(s, \pi_n)$ over $R$ replications of the same computational time measurement. To reduce $\text{RSD}_R(s, \pi_n)$, the operating system (Debian Linux 12.4 with "processor affinity for RT-tasks" based on kernel version 6.1.0-17) has been configured to avoid most of the OS's noises, a disruption may however occur due to potential non-maskable interruptions and unavoidable waiting times for external events or resource availability.

The values of $\text{RSD}_R(s,\ \pi_n)$ are not detailed for the sake of conciseness. $R$ has been chosen so that $\text{RSD}_R(s,\ \pi_n) \leq 5\%$ in "almost all cases" (i.e. for the large majority of the sorts and values of $n$). However, due to above-mentioned external variations, this target is not reached for all sorts with $n = 8$ on i7 and FPGA as well as for quick-sort and tim-sort with $n = 16$ on i7.

Nonetheless, it should be noted that the average $\text{RSD}_R(s, \pi_n)$ over all sorts $s$ and all permutations $\pi_n$ is approximately equal to 1.2% which is far less than the expected limit of 5%. In our design of experiments, $\text{RSD}_P(s)$ is computed for each sort $s$ to assess the representativeness of its average computational time.

Actually, the stability of sorts in terms of computational time is assumed significant if $\text{RSD}_P(s) \leq 5\%$, otherwise statistical tests are performed. Moreover, in order to rank the sorts, confidence intervals $\text{CI}(s)$ are computed as follows:

$$\text{CI}(s) = \left[\mu_P(s) - 1.96 \cdot \frac{\sigma_P(s)}{\sqrt{P}}, \mu_P(s) + 1.96 \cdot \frac{\sigma_P(s)}{\sqrt{P}}\right]. \quad (2)$$

In addition to the focus on a single sort, boxplots [41] are useful tools for visualizing and comparing distributions of computational time measurements on the same scale. Specifically, boxplots allow us to compare and analyze the stability of different algorithms.

Standard boxplots are usually based on five values that summarize the data ($Q0$ (min), $Q1$ (first quartile), $Q2$ (median), $Q3$ (third quartile), and $Q4$ (max)) for the studied population (here, 47 permutations). As usual, $IQR$ is also defined as the difference between the third and first quartiles. All observations above $Q3 + 1.5 \times IQR$ or below $Q1 - 1.5 \times IQR$ are considered as outliers.

On the basis of the number of outliers # outliers, this allows us to compute the percentage of outliers as follows:

$$\#\ \text{outliers}(s) = 100 \times \frac{\#\text{outliers}(s)}{P}. \quad (3)$$

Upper outliers (denoted by #$^+$) are defined as all observations above $Q3 + 1.5 \times IQR$. Similarly, %$^+$outliers is the percentage of upper outliers. Each set of $P$ experiments (on $P$ distinct

**Table 6.** Upper outliers percentages %⁺ outliers on i7

| size | tim-sort | merge-sort | heap-sort | shell-sort | quick-sort |
|---|---|---|---|---|---|
| 8 | 2.13 | 0.00 | 0.00 | 0.00 | 6.38 |
| 16 | 4.26 | 8.51 | 0.00 | 0.00 | 10.64 |
| 32 | 8.51 | 2.13 | 2.13 | 8.51 | 12.77 |
| 64 | 0.00 | 0.00 | 6.38 | 4.26 | 12.77 |
| 128 | 6.38 | 0.00 | 6.38 | 8.51 | 12.77 |
| 256 | 6.38 | 0.00 | 6.38 | 2.13 | 12.77 |
| 512 | 0.00 | 0.00 | 0.00 | 6.38 | 12.77 |
| 1024 | 2.13 | 0.00 | 0.00 | 4.26 | 12.77 |
| 2048 | 2.13 | 0.00 | 0.00 | 0.00 | 12.77 |
| 4096 | 0.00 | 0.00 | 0.00 | 4.26 | 12.77 |

permutations) of a sort $s$ may be represented by a boxplot and summarized by $Q0(s)$ to $Q4(s)$.

Supplemented by the outliers, $\sigma_P(s)$, $\mu_P(s)$ and $\text{RSD}_P(s)$, it is possible to obtain a precise view of the temporal stability of a sort. $\text{RSD}_P(s) < 5\%$ and/or a small value of $IQR(s)$ suggest(s) that the sort $s$ is stable, this is usually confirmed by a small value of %outliers$(s)$.

On the contrary, $\text{RSD}_P(s) > 5\%$ and/or a large value of $IQR(s)$ and/or a significant difference between $Q2(s)$ and $\mu_P(s)$ suggest(s) that the sort is not stable. In the context of real-time applications, particular attention should be payed to the percentage of upper outliers (%⁺outliers) since it provides information on worst-case computational times of a given sort.

In other words, depending on input data this sort may reach computational times that are not compatible with the targeted time-constraints.

Moreover, by grouping several boxplots (one per sort) in the same plot, it is possible to visually compare the performance and temporal stability of the sorts. In addition to $\mu_P(s)$ and $\text{RSD}_P(s)$, a significant variation of $Q2(s)$ between different distributions (i.e. sets of $P$ permutations, one set per sort $s$), suggests that the sorting times vary and it is possible to rank the sorts.

In contrast, if the medians are quite similar across several distributions, $\sigma_P(s)$, $\mu_P(s)$ and/or boxplots are not sufficient to rank the sorts and statistical tests are required. Consequently, it should have been interesting to present boxplots and statistical tests for all sorts on all target architectures, however for the sake of conciseness, numerical results have been reduced to $\mu_P(s)$, $\sigma_P(s)$, $\text{RSD}_P(s)$ and %outliers$(s)$ when/where sufficient. The results of statistical tests are also summarized when needed.

In our design of experiments, the following statistical tests are performed: First of all, to assess the normality of the data distributions, the Shapiro-Wilk test is conducted on each set of $P$ experiments and each value of $n$. As a result of these tests, it appears that the data deviate significantly from a normal distribution, consequently nonparametric tests must be used.

So, in a second step, the Kruskal-Wallis test is employed to examine the overall differences in computational times among the five sorts on each platform. Subsequently, to refine the results and rank the sorts as far as statistically possible, pairwise comparisons using Wilcoxon tests are conducted to identify specific algorithm pairs that exhibited significant differences.

For a given architecture, if all tests are performed on the $\eta = |\mathcal{N}|$ values of $n$, this leads to $\eta \times \Sigma$ Shapiro-Wilk tests, $\eta$ Kruskal-Wallis tests and $\eta \times (\Sigma \times (\Sigma - 1))/2$ Wilcoxon tests. This makes a total of – at most – 160 statistical tests (with $\eta = 10$

**Table 7.** Average and standard deviation of computational times on FPGA

| size/us | tim-sort | merge-sort | heap-sort | shell-sort | quick-sort |
|---|---|---|---|---|---|
| **8** | 18.12 (3.47) | 18.93 (3.49) | 17.99 (3.49) | 17.88 (3.48) | 18.68 (3.51) |
| **16** | 22.07 (3.47) | 23.60 (3.48) | 21.85 (3.48) | 22.25 (3.48) | 23.22 (3.49) |
| **32** | 31.18 (3.47) | 34.48 (3.47) | 31.10 (3.48) | 33.25 (3.48) | 34.30 (3.48) |
| **64** | 50.89 (3.47) | 58.46 (3.48) | 52.78 (3.47) | 60.24 (3.48) | 62.80 (3.48) |
| **128** | 94.50 (3.47) | 111.88 (3.49) | 102.67 (3.49) | 126.30 (3.48) | 137.90 (3.48) |
| **256** | 189.00 (3.47) | 231.30 (3.48) | 228.57 (3.48) | 293.55 (3.48) | 351.20 (3.48) |
| **512** | 393.17 (3.47) | 482.50 (3.49) | 466.05 (3.48) | 668.20 (3.48) | 1001.30 (3.48) |
| **1024** | 832.85 (3.47) | 1031.50 (3.48) | 1015.50 (3.47) | 1625.10 (3.56) | 3121.00 (3.49) |
| **2048** | 1769.50 (3.47) | 2211.50 (3.47) | 2227.10 (3.49) | 4027.67 (3.48) | 10660.00 (3.48) |
| **4096** | 3756.00 (3.47) | 4734.90 (3.48) | 4848.50 (3.48) | 9756.40 (3.48) | 38507.00 (3.49) |

and $\Sigma = 5$) performed via several scripts written in R language [23].

The effective number of pairwise comparisons can be reduced if it is guided by the ranking of sorts by $\mu_P$. All p-values are adjusted using the Bonferroni correction method so as to obtain an overall $\alpha$ level set to 5%.

For a given architecture (i7 or FPGA) and a value of $n$, our DOE follows several steps described in a simplified way as a pseudo-code in Algorithm 1, supplemented by some comments about main lines of the algorithm.

As shown in previous paragraphs, some steps of Algorithm 1 are dedicated to check the stability of the computational environment while others are used to assess the temporal stability of each sort and the others deal with the total or partial ranking of the sorts on the basis of their computational times. Finally, it is important to note that the resulting ranking in $\mathcal{S}'$ is not necessarily the same when $n$ varies.

### 3.2 Performances Study of Sorting Algorithms on i7

This section illustrates our DOE to compare and analyze the sorts on an Intel i7-9850H @ 2.60 GHz. All codes are written in C language and compiled via gcc version 12.2.0-14 (on Linux Debian 12.4) with the O3 optimization flag turned on. The sorts are evaluated in terms of computational times. They are then rated on the basis of their temporal stability.

In more details, the first step of our DOE computes $\text{RSD}_R(s)$ to check for external variations from the computational environment (lines 5 to 8 of Algorithm 1). Then, subsection 3.2.1 illustrates the second step of Algorithm 1 (lines 10 to 12). In a similar way, subsection 3.2.2 follows the third step of Algorithm 1 (lines 13 to 17). Finally, subsection 3.2.3 is dedicated on the last step of Algorithm 1 (lines 18 to 20).

We detail each of the steps 2 to 4 of Algorithm 1 in a separate subsection to illustrate how it works and "its" results in several tables, but in reality the steps follow one another in the algorithm to lead if necessary to statistical tests and temporal stability analysis.

#### 3.2.1 Average and Standard-Deviation of Computational Times on i7

To illustrate the second step of Algorithm 1 (lines 10 to 12) Table 2 displays the average and

**Table 8.** $\text{RSD}_P(s)$ of sorting algorithms on FPGA

| size% | tim-sort | merge-sort | heap-sort | shell-sort | quick-sort |
|---|---|---|---|---|---|
| **8** | 19.15 | 18.44 | 19.40 | 19.43 | 18.79 |
| **16** | 15.72 | 14.75 | 15.91 | 15.61 | 15.03 |
| **32** | 11.13 | 10.07 | 11.10 | 10.46 | 10.14 |
| **64** | 6.82 | 5.94 | 6.57 | 5.76 | 5.54 |
| **128** | 3.67 | 3.10 | 3.38 | 2.75 | 2.52 |
| **256** | 1.83 | 1.50 | 1.51 | 1.18 | 0.99 |
| **512** | 0.88 | 0.44 | 0.78 | 0.52 | 0.34 |
| **1024** | 0.42 | 0.34 | 0.40 | 0.22 | 0.11 |
| **2048** | 0.20 | 0.16 | 0.16 | 0.09 | 0.03 |
| **4096** | 0.09 | 0.07 | 0.07 | 0.04 | 0.00 |

standard deviation of computational times ($\mu_P(s)$ and $\sigma_P(s)$) whereas Table 3 shows the relative standard deviation $\text{RSD}_P(s)$. In this subsection, the objective is to rank the sorts according to $\mu_P(s)$ while considering $\text{RSD}_P(s)$ and $\text{CI}(s)$ as in the second step of Algorithm 1.

In this step, $\text{RSD}_P(s)$ and $\text{CI}(s)$ are not used to assess the temporal stability, but respectively to check whether statistical tests are needed or whether it is possible to obtain a total or partial ranking of the sorts. On i7, these measurements lead to the ranking of the sorts presented in Table 4.

In this table, the sorts are ranked from left (first/best/fastest sort) to right (last/worst/slowest sort) according to their $\mu_P(s)$. The $\prec$ symbol indicates a strict ranking, with no intersection of confidence intervals whereas $?$ denotes that the confidence intervals overlap.

At this second step of Algorithm 1, based on $\mu_P(s)$ and $\text{CI}(s)$, Table 2 and Table 4 show that for $n < 64$, tim-sort outperforms other sorts in terms of $\mu_P$. Tim-sort, however, has a higher $\text{RSD}_P(s)$ than merge-sort, heap-sort, and shell-sort, which is not a good clue – at this step – of its temporal stability.

Similarly, for $n > 512$, heap-sort outperforms other sorts in terms of $\mu_P$. In between, the rankings are not clearly established due to intersections of the confidence intervals and it is challenging to draw a significant conclusion regarding the best algorithm.

For $n \geq 16$, it should be noted in Table 3 that $\text{RSD}_P(s)$ for heap-sort is less than 5% (or slightly greater than 5%) and $Q2(S) \approx \mu_P(s)$ (the maximum gap between $Q2(S)$ and $\mu_P(s)$ is equal to 2%) leading Algorithm 1 to line 12. However, since other sorts have higher values of $\text{RSD}_P(s)$ and a gap between $Q2(S)$ and $\mu_P(s)$ greater than 5%, statistical tests will be used for pairwise comparisons in the next step.

As a detail of step 2 of Algorithm 1, we first present a ranking based on the average and the standard deviation of computational times because this is what is conventionally used in the literature, warning however about the intersections between $\text{CI}(s)$ that do not allow to establish a total ranking of sorts.

At this step, the results of quick-sort should be viewed with great caution because its $\text{RSD}_P(s)$ is really high and there is a very large gap between $Q2(s)$ and $\mu_P(s)$. As for the other sorts, since the $\text{RSD}_P(s)$ values are in large majority greater than 5%, statistical tests are used in a next step to confirm or refute the rankings.

With the previous conclusions in mind, we can calculate the relative gains in terms of $\mu_P(s)$, even if the values of $\text{RSD}_P(s)$ and $\text{CI}(s)$ temper these results. For $n < 64$, tim-sort is 63-70% faster than merge-sort, 46-59% faster than heap-sort,

**Table 9.** Ranking of sorts using $\mu_P(s)$ and CI$(s)$ on FPGA

| **size** | $\mathcal{S}'$ | | | | | | | | | **RSD**$_P$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 8 | shell | $?$ | heap | $?$ | tim | $?$ | quick | $?$ | merge | $> 5\%$ |
| 16 | heap | $?$ | tim | $?$ | shell | $?$ | quick | $?$ | merge | $> 5\%$ |
| 32 | heap | $?$ | tim | $\preceq$ | shell | $?$ | quick | $?$ | merge | $> 5\%$ |
| 64 | tim | $?$ | heap | $\prec$ | merge | $?$ | shell | $\preceq$ | quick | $> 5\%$ |
| 128 | tim | $\prec$ | heap | $\prec$ | merge | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |
| 256 | tim | $\prec$ | heap | $\preceq$ | merge | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |
| 512 | tim | $\prec$ | heap | $\prec$ | merge | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |
| 1024 | tim | $\prec$ | heap | $\prec$ | merge | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |
| 2048 | tim | $\prec$ | merge | $\prec$ | heap | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |
| 4096 | tim | $\prec$ | merge | $\prec$ | heap | $\prec$ | shell | $\prec$ | quick | $\leq 5\%$ |

24-32% faster than shell-sort and 36-43% faster than quick-sort.

Furthermore, for $n = 64$ and $n = 128$, shell-sort is 39-46% faster than merge-sort, 35-40% faster than heap-sort, 20-28% faster than tim-sort, and 20-25% faster than quick-sort.

Finally, if $n > 1024$, heap-sort is 56-80% faster than quick-sort, 41-45% faster than shell-sort, 28-32% faster than merge-sort, and 27-33% faster than tim-sort, as mentioned in Table 2. As a conclusion of this step, the rankings of the sorts – at least for $n \geq 64$ – need validations through statistical tests in next step.

### 3.2.2 Statistical-Tests-Based Ranking on i7

This subsection is based on the third step of Algorithm 1 (lines 13 to 17) which leads to the ranking presented in Table 5. This step is based on nonparametric statistical tests that base their ranking on medians (i.e. $Q2(s)$) and not on averages (i.e. $\mu_P(s)$), unlike the ranking presented in Table 4. This can lead to different rankings if there is a large gap between $Q2(s)$ and $\mu_P(s)$.

This is the case, for example, with the quick-sort, which is ranked first by statistical tests while its results in terms of RSD$_P(s)$ are disastrous as shown in Table 3, relating to previous step of Algorithm 1.

In fact, the quick-sort is tagged as highly unstable by the Algorithm 1 and excluded from the ranking (since for $n > 8$, RSD$_P(s) \gg 5\%$ and there is a very large gap between $Q2(s)$ and $\mu_P(s)$). This explains the use of RSD$_P(s)$ as early as step 2 of the algorithm.

When considering tim-sort, merge-sort and shell-sort, Table 3, shows that their RSD$_P(s)$ is also greater than 5% but the gap between $Q2(s)$ and $\mu_P(s)$ is less that 12% (i.e. far less than the gap for quick-sort) so these sorts are included in current and next step of Algorithm 1.

The results of statistical tests summarized in Table 5 confirm the rankings obtained in Table 4 for $n = 8$, $n = 16$ and $n = 32$. When discarding quick-sort from the rankings, statistical tests confirm the rankings obtained in Table 4 for $n = 64$, $n = 128$, $n = 256$, $n = 2048$ and $n = 4096$ (i.e. respectively shell $\prec$ tim $\prec$ heap $\prec$ merge, shell $\prec$ tim $\prec$ heap $\prec$ merge, tim $\prec$ merge $\prec$ shell $\preceq$ heap, heap $\prec$ merge $\prec$ tim $\prec$ shell and heap $\prec$ merge $\prec$ tim $\prec$ shell). However, for $n = 512$ and $n = 1024$ the rankings are different when comparing Table 5 to Table 4.

Nevertheless, heap-sort is ranked first for $n = 1024$ in the two tables and lead to a total pairwise ranking (denoted by $\prec$ in Table 5). On the contrary, the rankings are different for $n = 512$ in these two tables. It is interesting to note that this ranking, in Table 4, is tagged as not significant

since there are intersections of CI$(s)$ (denoted by $?$ in this table).

As a first conclusion of this step, it should be mentioned that the first rankings of the sorts using $\mu_P(s)$ and CI$(s)$ in Table 4 provide us with very accurate rankings for almost all values of $n$, it has been able to identify the fastest sort in 9 out of 10 cases while discarding the worst one (i.e. quick-sort).

As a second conclusion, this current test, based on statistical test confirms and refines all rankings obtained in previous step, leading to the following choices: Tim-sort is the fastest sort for $8 \leq n \leq 32$ and $n = 256$, shell-sort is the faster sort for $n = 64$ and $n = 128$, heap-sort is the fastest sort for $n \geq 1024$, it is not more than 95% sure that merge-sort is faster than heap-sort for $n = 512$ in this latter case their temporal stabililies is used in next step to refine the choice for $n = 512$.

### 3.2.3 Temporal Stability of Sorting Algorithms on i7

This subsection is based on the last step of Algorithm 1 (lines 18 to 20) which leads to the results presented in Table 6. First of all, it's worth mentioning that none of these sorts are ideally stable and there are "acceptable" temporal variations. These variations can be bounded by upper values using the upper bound of CI$(s)$ or Q3$(s)$. In this paper, this bound is set to $Q3(s)+1.5\times IQR(s)$, which is the standard definition of upper outliers.

Consequently – at this step of algorithm Algorithm 1 (line 19) – each sort $s$ such as %$^+$outliers$(s) > 0$ should be considered as non temporally stable and discarded from the final ranking of the sorts. It is also important to bound the relative dispersion of computational times around the average and this is exactly the definition of RSD$_P(s)$.

Consequently, each sort $s$ such as RSD$_P(s) > 5\%$ should be considered as non temporally stable and discarded from the while ranking the sorts, either in this current step or in previous steps of Algorithm 1 (line 12 or line 17). Finally, if there are too many variations from the computational environment (Algorithm 1, line 20), it is possible to discard each sort $s$ such that badRSDR$[s]$ = true.

As previously mentioned, it should have been interesting to represent the results using boxplots, however due to lack of space the comparisons are exclusively based on numerical values RSD$_R(s, \pi_n)$, CI$(s)$, RSD$_P(s)$ and %$^+$outliers$(s)$. To this end, RSD$_P(s)$ are given in Table 3 and (%$^+$outliers) are given in Table 6.

Before finalizing the ranking, it is important to mention that in our target application, a duration of less than one microsecond is considered negligible, consequently there no real challenge on temporal stability when considering the sorts for $n \leq 64$ (excepted for quick-sort as previously mentioned).

So for $n \leq 64$ any sort $s$ might be considered as temporally stable, even if %$^+$outliers$(s) > 0$ since its overall duration, including the upper bound of CI$(s)$ or Q3$(s)$ are within the tolerance ranges of one microsecond. As a general comment, it is noticeable that, in Table 3, RSD$_P(s)$ are greater than 1% and even mainly greater than 5%.

Consequently, a good choice might be the fastest sort or the most temporally stable one, that is either tim-sort for $8 \leq n \leq 32$ and shell-sort for $n = 64$ or (fastest sorts) heap-sort for $8 \leq n \leq 32$ and tim-sort for $n = 64$ (most temporally stable sorts) or another combination of the sorts. For $64 \leq n \leq 256$, the best sort in terms of %$^+$outliers$(s)$ is merge-sort whereas the best sort in terms of RSD$_P(s)$ is heap-sort (even if RSD$_P(s)$ for merge-sort are very close to those of heap-sort).

In the same time, for $64 \leq n \leq 256$, the fastest sorts are shell-sort and team-sort. For $n \geq 512$, the best sorts in terms of %$^+$outliers$(s)$ are merge-sort and heap-sort whereas terms of RSD$_P(s)$ is clearly heap-sort. In the same time, for $n \geq 512$, the fastest sorts are also merge-sort and heap-sort. Consequently the best choice for $n \geq 512$ is the heap-sort.

At the end of all the steps of Algorithm 1 we can derive a definitive conclusion regarding the sorting method(s) to be employed for our avionics application on I7. It is clear that the "best sort" on i7 depends on $n$ but also on the criterion to be minimized, i.e. either the computational time

or the temporal stability. It is clear that the best choice considering both criteria for $n \geq 512$ is the heap-sort. For $64 \leq n \leq 256$, the choice is not so easy whereas for $n \leq 64$ the computational times and the relating variations are so small that any sort is acceptable on i7.

### 3.3 Performances Study of Sorting Algorithms on FPGA

As for i7 in section 3.2, this section illustrates our DOE to compare and analyze the sorts on FPGA. All codes are written in C language and the optimized hardware implementation is generated using HLS directives (loop unrolling, loop pipelining, input/output interface). Vivado is used for synthesis and running the VHDL architecture. The sorts are evaluated in terms of computational times and temporal stability.

On FPGA our DOE follows the steps of Algorithm 1: The first step of our DOE computes $\text{RSD}_R(s)$ to check for external variations from the computational environment (lines 5 to 8 of Algorithm 1). Then, subsection 3.3.1 illustrates the second step of Algorithm 1 (lines 10 to 12).

Contrary to i7, the values of $\text{RSD}_P(s)$ on FPGA are such that there is no need to use statistical tests and the third step of Algorithm 1 is "skipped". Finally, subsection 3.3.2 is dedicated to the last step of Algorithm 1 (lines 18 to 20). In the following subsections, steps 2 and 4 of Algorithm 1 are detailed to illustrate how it works and "its" results in several tables.

#### 3.3.1 Average and Standard Deviation of Computational Times on FPGA

To illustrate the second step of Algorithm 1, Table 7 displays the average and standard deviation of computational times ($\mu_P(s)$ and $\sigma_P(s)$) whereas Table 8 shows the relative standard deviation $\text{RSD}_P(s)$.

In this subsection, the objective is to rank the sorts according to $\mu_P(s)$ while considering $\text{RSD}_P(s)$ and $\text{CI}(s)$ as in the second step of Algorithm 1. In this step, $\text{RSD}_P(s)$ and $\text{CI}(s)$ are used to check whether statistical tests are needed or whether it is possible to obtain a total or partial ranking of the sorts. On FPGA, these measurements lead to the ranking of the sorts presented in Table 9.

In this table, the sorts are ranked from left (first/best/fastest sort) to right (last/worst/slowest sort) according to their $\mu_P(s)$. The $\prec$ symbol indicates a strict ranking, with no intersection of confidence intervals whereas $?$ denotes that the confidence intervals overlap. In addition, $\preceq$ indicates that the confidence intervals are contiguous within a range of 1 us, meaning they are "nearly disjoint".

At this second step of Algorithm 1, based on $\mu_P(s)$ and $\text{CI}(s)$, Table 7 and Table 9 show that – in terms of $\mu_P$ – for $n = 8$ shell-sort outperforms other sorts, for $n = 16$ or $n = 32$ heap-sort outperforms other sorts, and for $n = 64$ tim-sort outperforms other sorts. However in previous rankings, these "bests sorts" are tagged as not significant since there are intersections of $\text{CI}(s)$ (denoted by $?$ in Table 9).

It is worth noting that for $n \leq 64$, $\text{RSD}_P(s) > 5\%$ for all sorts, as summarized in Table 9, last column. Before finalizing the ranking, it is important to mention that in our FPGA, $\sigma_P(s)$ is mainly due to hardware perturbations and appears as "almost constant" in Table 7 for all $n$.

This explains that $\text{RSD}_P(s)$ is decreasing as a function of $n$. This also means that hardware perturbations are too high to rank the sort for $n \leq 64$, which is confirmed by $\text{RSD}_P(s) > 5\%$ for all sorts. Therefore, these rankings should be viewed with caution and the hardware perturbations are such that statistical tests would not refine the results.

In Table 9, the results are completely different for $n > 64$ and for all sorts since all pairwise comparisons of sorts are such that confidence intervals of sorts do not intersect (excepted for heap-sort and merge-sort for $n = 256$ where confidence intervals are contiguous within a range of 1 us).

Moreover, for $n > 64$ tim-sort is ranked first and this is confirmed by the fact that $\text{RSD}_P(s) < 5\%$ for all sorts, with no need for statistical tests (since $Q2(S) \approx \mu_P(s)$ is also verified), leading Algorithm 1 to line 12. With the previous conclusions in mind, we can calculate the relative gains in terms of

$\mu_P(s)$, even if the values of RSD$_P(s)$ and CI$(s)$ temper these results.

For $n < 64$, heap-sort is 1.01x-1.12x faster than other algorithms. Furthermore, for $n = 4096$, the results show that tim-sort has an average computational time of 3756.00 us, while merge-sort, heap-sort, shell-sort, and quick-sort have average computational times of 4734.90 us, 4848.50 us, 9756.40 us, and 38507.00 us respectively. When comparing tim-sort with the other sorts, the results show that tim-sort is 1.16x-1.21x, 1.08x-1.23x, and 1.25x-1.61x faster than merge-sort, heap-sort, and shell-sort respectively, if $n > 64$.

As a conclusion of this step, the rankings of the sorts on the basis of computational times on FPGA for $n \leq 64$ need to be considered with caution due to the hardware perturbations. Indeed, $\sigma_P(s)$ is almost constant for all sort $s$ and all $n$, which is due to the synchronous design of the FPGA. Therefore, the average temporal variation does not depend on $n$ (with $n \leq 4096$).

Moreover, Table 9 shows that for $n > 64$, tim-sort is the fastest sort in terms of average computational time. Additionally, there is no intersection of confidence intervals while considering pairwise comparisons of the sorts.

#### 3.3.2 Temporal Stability of Sorting Algorithms on FPGA

This subsection is based on the last step of Algorithm 1 (lines 18 to 20). First of all, it's worth mentioning that none of these sorts are ideally stable and there are "acceptable" temporal variations if $n \leq 64$.

Consequently, each sort $s$ such as RSD$_P(s) > 5\%$ should be considered as non temporally stable and discarded from the while ranking the sorts, either in this current step or in previous steps of Algorithm 1.

Finally, if there are too many variations from the computational environment (Algorithm 1, line 20), it is possible to discard each sort $s$ such that badRSDR$[s]$ = true. Due to the synchronous design of the FPGA and the above-mentioned hardware perturbations, the results of this step can be summarized in a few words.

For $n \leq 64$, RSD$_P(s) > 5\%$ and it is not possible to precisely measure the intrinsic temporal stability of the sorts. For $n > 64$, RSD$_P(s) < 5\%$ (as shown in Table 8) and CI$(s)$ are such that the sorts are considered as temporally stable on FPGA with a constant maximum value for $\sigma_P(s)$ and RSD$_R(s, \pi_n) < 5\%$.

At the end of all the steps of Algorithm 1 we can derive a definitive conclusion regarding the sorting method(s) to be employed for our avionics application on FPGA. It is clear that the "best sort" on FPGA depends on $n$ but the temporal stability is induced by the synchronous design of the FPGA.

This explains that for $n \leq 64$ it is almost impossible to distinguish the sorts from the point of view of computational times as well as from their temporal stability. This also explains the fact that it is possible to clearly distinguish (i.e. with small temporal variations, measured by $\sigma_P(s)$) the "best sort" from the point of view of computational times for $n > 64$, i.e. tim-sort. Moreover, for $n > 64$, the sorts are also undistinguishable from the point of view of their temporal stability.

### 3.4 Comparison of the Computational Platforms

Despite a much lower frequency, computational times on FPGA are "respectable" compared to those on i7 because Xilink's tool is able to extract the parallelism of the algorithms by means of the optimizations introduced via HLS directives. However, a decrease in terms of average and standard deviation of computational times leads to an increase in resource utilization on FPGA.

Additionally, it is worth noting that the temporal stability of the hardware implementation on FPGA is much better than that on i7, when considering the relative variations given by RSD$_P(s)$ or when considering CI$(s)$.

For conclusion, FPGA provides a better temporal stability than Intel i7 but sorts on FPGA are slower than on i7, even if FPGA offers high performance in terms of parallelism. On the contrary, i7 leads to worse performance than FPGA in terms of temporal stability, when considering RSD$_P(s)$.

However, the sorts run much faster on i7 than on FPGA and it is possible to act on worst-case computational times and limit the time variations by reducing the upper outliers (#$^+$outliers$(s)$) thanks to dedicated configuration of the operating system, based on "processor affinity for RT-tasks".

The choice of the "best sort" from the points of view of computational time and temporal stability is clear on FPGA and tim-sort appears to be the ranked first. The same choice on i7 is not so easy and depends on $n$ but also on the criterion to be minimized, i.e. either the computational time or the temporal stability. It appears that the best choice considering both criteria for $n \geq 512$ is the heap-sort.

For $64 \leq n \leq 256$, the choice is not so easy whereas for $n \leq 64$ the computational times and the relating variations are so small that any sort is acceptable on i7 when considering the maximum time-constraints of our target application.

## 4 Conclusions

In this paper, we presented a review of different works using sorting algorithms on Intel i7 and FPGA architectures. To conduct our study, a high-level description of the sorting algorithms is used on FPGA. Our evaluation of various sorts provides valuable insights into their performance and stability, which can guide the selection of suitable algorithms for real-time decision support applications in the avionics industry.

Indeed, a stable sort is particularly useful for real-time targeted applications. In the context of real-time applications, particular attention should be payed to the percentage of upper outliers since it provides information on worst-case computational times of a given sort. In other words, depending on input data this sort may reach computational times that are not compatible with the targeted time-constraints.

The obtained results show that it is difficult to choose the best algorithms on Intel i7, on the contrary tim-sort have a better performance on FPGA for $n \geq 64$. We concluded that the FPGA provides a better performance in terms of temporal stability. We show experimentally that the same sorting algorithms are not ranked in the same way on two different architectures.

Additionally, the calculation of the average and standard deviation of computational times may not be sufficient – depending on the target architecture – to rank these sorts in a statistically representative manner. Similarly, the stability of sorting algorithms may vary from one architecture to another one and/or depending on the size of the data to be sorted. Consequently, we are working on combinations of sorting algorithms to propose a hybrid sort that offers the best possible performance both in terms of computation time and temporal stability.

As future work, we plan to use the hardware version of "the best sorting algorithm" in our targeted avionics decision support system [35, 40, 36]. The present work is also inspired by other researches dedicated to the optimization of matching and scheduling on heterogeneous CPU/FPGA architectures [40] where efficient sorts are required.

## References

1. **Abdelrasoul, M., Shaban, A. S., Abdel-Kader, H. (2021).** Based hardware accelerator for sorting data. Proceedings of the 9th International Japan-Africa Conference on Electronics, Communications, and Computations, pp. 57–60. DOI: 10.1109/JAC-ECC54461.2021.9691432.
2. **Abdelrasoul, M., Shaban, A. S., Abdel-Kader, H. (2021).** Index and sort algorithm based on fpga for sorting data. Proceedings of the 9th International Japan-Africa Conference on Electronics, Communications, and Computations, pp. 61–64. DOI: 10.1109/JAC-ECC54461.2021.9691445.
3. **Adam, G. K. (2022).** Co-design of multicore hardware and multithreaded software for thread performance assessment on an FPGA. Computers, Vol. 11, No. 5, pp. 76. DOI: 10.3390/computers11050076.

4. **Al-Kharabsheh, K. S., Al-Turani, I. M., Al-Turani, A. M. I., Zanoon, N. I. (2013).** Review on sorting algorithms a comparative study. International Journal of Computer Science and Security, Vol. 7, No. 3, pp. 120–126.
5. **Almomany, A., Ayyad, W. R., Jarrah, A. (2022).** Optimized implementation of an improved KNN classification algorithm using intel FPGA platform: COVID-19 case study. Journal of King Saud University - Computer and Information Sciences, Vol. 34, No. 6, pp. 3815–3827. DOI: 10.1016/j.jksuci.2022.04.006.
6. **Auger, N., Jugé, V., Nicaud, C., Pivoteau, C. (2020).** Analysis of timsort algorithm. Laboratoire D'Informatique Gaspard-Monge.
7. **Bai, Y., Ahmed, S. Z., Granado, B. (2014).** Fast and power efficient heapsort IP for image compression application. Proceedings of the IEEE 22nd Annual International Symposium on Field-Programmable Custom Computing Machines, pp. 237–237. DOI: 10.1109/FCCM.2014.72.
8. **Bai, Y., Ahmed, S. Z., Granado, B. (2014).** A power-efficient adaptive heapsort for fpga-based image coding application (abstract only). Proceedings of the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 247. DOI: 10.1145/2554688.2554746.
9. **Ben-Jmaa, Y., Ben-Atitallah, R., Duvivier, D., Ben-Jemaa, M. (2019).** A comparative study of sorting algorithms with FPGA acceleration by high level synthesis. Computación y Sistemas, Vol. 23, No. 1, pp. 213–230. DOI: 10.13053/cys-23-1-2999.
10. **Ben-Jmaa, Y., Duvivier, D., Abid, M. (2021).** Sorting algorithms on ARM cortex A9 processor. Lecture Notes in Networks and Systems, pp. 355–366. DOI: 10.1007/978-3-030-75078-7_36.
11. **Ben-Jmaa, Y., Duvivier, D., Abid, M. (2022).** ARM vs FPGA: Comparative analysis of sorting algorithms. Lecture Notes in Networks and Systems, pp. 275–287. DOI: 10.1007/978-3-030-99619-2_27.
12. **Bramas, B. (2021).** A fast vectorized sorting implementation based on the ARM scalable vector extension (SVE). PeerJ Computer Science, Vol. 7, pp. e769. DOI: 10.7717/peerj-cs.769.
13. **Chen, C., da-Silva, B., Chen, R., Li, S., Li, J., Liu, C. (2022).** Evaluation of fast sample entropy algorithms on FPGAs: From performance to energy efficiency. Entropy, Vol. 24, No. 9, pp. 1177. DOI: 10.3390/e24091177.
14. **Chen, H., Madaminov, S., Ferdman, M., Milder, P. (2020).** FPGA-accelerated samplesort for large data sets. Proceedings of the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, ACM, pp. 222–232. DOI: 10.1145/3373087.3375304.
15. **Chen, W., Li, W., Yu, F. (2020).** A hybrid pipelined architecture for high performance top-k sorting on FPGA. IEEE Transactions on Circuits and Systems II: Express Briefs, Vol. 67, No. 8, pp. 1449–1453. DOI: 10.1109/TCSII.2019.2938892.
16. **Coussy, P., Gajski, D. D., Meredith, M., Takach, A. (2009).** An introduction to high-level synthesis. IEEE Design and Test of Computers, Vol. 26, No. 4, pp. 8–17. DOI: 10.1109/MDT.2009.69.
17. **Diallo, A., Zopf, M., Fürnkranz, J. (2020).** Permutation learning via lehmer codes. 24th European Conference on Artificial Intelligence, IOS Press, pp. 1095–1102.
18. **Draz, H. H., Elashker, N. E., Mahmoud, M. M. A. (2023).** Optimized algorithms and hardware implementation of median filter for image processing. Circuits, Systems, and Signal Processing, Vol. 42, No. 9, pp. 5545–5558. DOI: 10.1007/s00034-023-02370-x.
19. **Esau-Taiwo, O., Christianah, A. O., Oluwatobi, A. N., Aderonke, K. A., Kehinde,**

**A. J. (2020).** Comparative study of two divide and conquer sorting algorithms: Quicksort and mergesort. Procedia Computer Science, Vol. 171, pp. 2532–2540. DOI: 10.1016/j.procs.2020.04.274.

20. **Fang, J., Mulder, Y. T. B., Hidders, J., Lee, J., Hofstee, H. P. (2019).** In-memory database acceleration on FPGAs: A survey. The VLDB Journal, Vol. 29, No. 1, pp. 33–59. DOI: 10.1007/s00778-019-00581-w.
21. **Gowanlock, M., Karsin, B. (2019).** A hybrid CPU/GPU approach for optimizing sorting throughput. Parallel Computing, Vol. 85, pp. 45–55. DOI: 10.1016/j.parco.2019.01.004.
22. **Hanafi, M. R., Faadhilah, M. A., Dwi-Putra, M. T., Pradeka, D. (2022).** Comparison analysis of bubble sort algorithm with tim sort algorithm sorting against the amount of data. Journal of Computer Engineering, Electronics and Information Technology, Vol. 1, No. 1, pp. 29–38. DOI: 10.17509/coelite.v1i1.43794.
23. **Ihaka, R., Gentleman, R. (1996).** R: A language for data analysis and graphics. Journal of Computational and Graphical Statistics, Vol. 5, No. 3, pp. 299–314. DOI: 10.1080/10618600.1996.10474713.
24. **Jayaraman, S., Zhang, B., Prasanna, V. (2022).** Hypersort: High-performance parallel sorting on HBM-enabled FPGA. Proceedings of the International Conference on Field-Programmable Technology, pp. 1–11. DOI: 10.1109/ICFPT56656.2022.9974209.
25. **Kirkeby, M. H., Krabben, T., Larsen, M., Mikkelsen, M. B., Petersen, T., Rosendahl, M., Schoeberl, M., Sundman, M. (2022).** Energy consumption and performance of heapsort in hardware and software. arXiv, pp. 1–3.
26. **Kobayashi, R., Miura, K., Fujita, N., Boku, T., Amagasa, T. (2021).** A sorting library for FPGA implementation in OpenCL programming. Proceedings of the 11th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies, pp. 1–6. DOI: 10.1145/3468044.3468054.
27. **Kobayashi, R., Miura, K., Fujita, N., Boku, T., Amagasa, T. (2022).** An open-source FPGA library for data sorting. Journal of Information Processing, Vol. 30, pp. 766–777. DOI: 10.2197/ipsjjip.30.766.
28. **Kristo, A., Vaidya, K., Cetintemel, U., Misra, S., Kraska, T. (2020).** The case for a learned sorting algorithm. Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 1001–1016. DOI: 10.1145/3318464.3389752.
29. **Kuzmin, D., Tkachenko, M., Nikolaienko, A. (2023).** Visualization and analysis of sorting algorithms. Prospective Directions of Scientific Research in Engineering and Agriculture, pp. 74–81. DOI: 10.46299/isg.2023.mono.tech.1.3.2.
30. **Lade, S., Patil, K., Chhadikar, N., Chaudhari, A. (2019).** Effective sorting using parallel computing. International Journal of Advanced Research in Computer Engineering and Technology.
31. **Lobo, J., Kuwelkar, S. (2020).** Performance analysis of merge sort algorithms. Proceedings of the International Conference on Electronics and Sustainable Communication Systems, pp. 110–115. DOI: 10.1109/ICESC48915.2020.9155623.
32. **Moghaddamfar, M., Färber, C., Lehner, W., May, N. (2020).** Comparative analysis of OpenCL and RTL for sort-merge primitives on FPGA. Proceedings of the 16th International Workshop on Data Management on New Hardware, pp. 1–7. DOI: 10.1145/3399666.3399897.
33. **Montesdeoca, G., Asanza, V., Chica, K., Peluffo-Ordóñez, D. H. (2022).** Analysis of sorting algorithms using a WSN and environmental pollution data based on FPGA. Proceedings of the International Conference on Applied Electronics, pp. 1–4. DOI: 10.1109/AE54730.2022.9920090.

34. **Muthavarapu, A. K., Biswas, J., Barai, M. (2022).** An efficient sorting algorithm for capacitor voltage balance of modular multilevel converter with space vector pulsewidth modulation. IEEE Transactions on Power Electronics, Vol. 37, No. 8, pp. 9254–9265. DOI: 10.1109/TPEL.2022.3160665.

35. **Nikolajevic, K. (2016).** Dynamic autonomous decision-support function for piloting a helicopter in emergency situations. Ph.D. thesis, Laboratoire d'Automatique de Mécanique et d'Informatique industrielles et Humaines, Université de Valenciennes et du Hainaut-Cambresis, France.

36. **Ollivier-Legeay, H., Cadi, A. A. E., Belanger, N., Duvivier, D. (2020).** A 4D augmented flight management system based on flight planning and trajectory generation merging. Lecture Notes in Computer Science, pp. 184–195. DOI: 10.1007/978-3-030-63486-5_21.

37. **Rahul, G., Sandeep, P., Latha, Y. L. M. (2020).** Quicksort algorithm—an empirical study. Advances in Intelligent Systems and Computing, pp. 387–401. DOI: 10.1007/978-981-15-1480-7_33.

38. **Shinyamada, K., Yamawaki, A. (2021).** Effect of sorting algorithms on high-level synthesized image processing hardware. Proceedings of the 8th International Conference on Intelligent Systems and Image Processing, pp. 16–20.

39. **Soomro, I., Ali, H., Lashari, H. N., Maitlo, A. (2021).** Performance analysis of heap sort and insertion sort algorithm. International Journal of Emerging Trends in Engineering Research, Vol. 9. DOI: 10.30534/ijeter/2021/08952021.

40. **Souissi, O., Atitallah, R. B., Duvivier, D., Artiba, A. (2013).** Optimization of matching and scheduling on heterogeneous CPU/FPGA architectures. IFAC Proceedings Volumes, Vol. 46, No. 9, pp. 1678–1683. DOI: 10.3182/20130619-3-ru-3018.00196.

41. **Streit, M., Gehlenborg, N. (2014).** Bar charts and box plots. Nature Methods, Vol. 11, No. 2, pp. 117–117. DOI: 10.1038/nmeth.2807.

42. **Sunny, S. P., Narayanan-M, P. (2022).** Parallel sorting based OS-CFAR implementation in FPGA. Oceans, pp. 1–8. DOI: 10.1109/OCEANSChennai45887.2022.9775472.

43. **Zhang, T., Rahimi-Azghadi, M., Lammie, C., Amirsoleimani, A., Genov, R. (2023).** Spike sorting algorithms and their efficient hardware implementation: A comprehensive survey. Journal of Neural Engineering, Vol. 20, No. 2. DOI: 10.1088/1741-2552/acc7cc.

44. **Zhang, Z., Li, J. (2023).** A review of artificial intelligence in embedded systems. Micromachines, Vol. 14, No. 5, pp. 897. DOI: 10.3390/mi14050897.

*Article received on 25/02/2024; accepted on 17/05/2024.*
**Corresponding author is Yomna Ben-Jmaa.*

ISSN 2007-9737

# Evaluating the Impact of Removing Low-relevance Features in Non-retrained Neural Networks

Uriel Corona-Bermúdez, Ricardo Menchaca-Méndez*, Rolando Menchaca-Méndez, Erendira Corona-Bermúdez

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

ucoronab@ipn.mx, {ric, rmen, ecoronab2020}@cic.ipn.mx

**Abstract.** Feature selection is a widely used technique to boost the efficiency of machine learning models, particularly when working with high-dimensional datasets. However, after reducing the feature space, we must retrain the model to measure the impact of the removed features. This can be inconvenient, especially when dealing with large datasets of thousands or millions of instances, as it leads to computationally expensive processes. To avoid the costly procedure of retraining, this study evaluates the impact of predicting using neural networks that have not been retrained after feature selection. We used two architectures that allow feature removal without affecting the architectural structure: FT-Transformers, which are capable of generating predictions even when certain features are excluded from the input, and Multi-layer Perceptrons, by pruning unused weights. These methods are compared against XGBoost, which requires retraining, on various tabular datasets. Our experiments demonstrate that the proposed approaches achieve competitive performance compared to retrained models, especially when the removal percentage is up to 20%. Notably, the proposed methods exhibit significantly faster evaluation times, particularly on large datasets. These methods offer a promising solution for efficiently applying feature removals, providing a favorable trade-off between performance and computational costs.

**Keywords.** Feature selection, transformers, pruning models, neural networks.

## 1 Introduction

Collecting data is an extensive process that corporations employ to extract knowledge and enhance process efficiency [15]. This has resulted in the creation of large databases, presenting an opportunity to apply machine learning and deep learning techniques, which thrive on substantial amounts of data for achieving remarkable results. Nevertheless, the collected data may contain several variables, causing the creation of high-dimensional datasets.

When dealing with high-dimensional information, many challenges and complications arise, such as an exponential increase in computational effort, large waste of space, poor visualization capabilities, and the inability of machine learning algorithms to manage this data [20, 18].

Even though including more variables may theoretically allow for the storage of more information, this may not be beneficial in practice. This is due to the higher likelihood of encountering noisy and redundant information in the dataset [20, 14]. A straightforward approach for dealing with high-dimensional data is to remove features that have repeated information or constant values.

This type of data cleansing is a natural part of the process. However, we may also need to reduce features that are still important to the

**Table 1.** Hyperparameter search space for models selected for comparison

| Model | Hyperparameter | Search set |
|---|---|---|
| FT-Transformer | Embedding dimension | {128, 256} |
| | Number of heads | {4, 8, 16, 32} |
| | Number of layers | {2, 3, 4, 5} |
| | Attention dropout | {0.3} |
| | Point-wise neural network dropout | {0.1} |
| | Learning rate | $\{10^{-4}\}$ |
| | Weight decay | $\{10^{-4}\}$ |
| MLP | Hidden layers | {3, 6} |
| | Hidden units | {512} |
| | Attention dropout | {0.3} |
| | Dropout | {0.2} |
| | Learning rate | $\{10^{-3}\}$ |
| | Weight decay | {0.1} |
| XGBoost | Maximum depth | {5, 10, 15, 20} |
| | Learning rate | {0.1} |

problem, but not as essential as others. To achieve this, it is common to employ a feature selection algorithm. These algorithms help us reduce the number of features to a desired size, but they use complex rules to determine which features to remove. This allows us to streamline the dataset while retaining the most relevant information. Evaluating the quality of a feature subset can be computationally expensive. The process typically involves the following steps:

1. Train an initial model using all available features to establish a baseline performance metric (e.g., accuracy).
2. Apply a feature selection algorithm to create a subset of the features. Once the relevant features are selected.
3. Train a second model using this reduced feature set, and finally.
4. Compare the performance of the two models. If the performance degradation between the two models is unacceptable, you must repeat the process.

This may involve trying a different feature selection algorithm or tuning the parameters of the current feature selection algorithm (e.g., changing the size of the generated subset). This iterative process of training models with different feature subsets to evaluate their quality can be computationally expensive.

The most prominent models for tabular data are tree-based algorithms, such as XGBoost, and neural networks [4, 9, 10, 13, 11, 7]. When dealing with large datasets, these models can take a significant amount of time to train. This poses a challenge when evaluating feature subsets generated by selection algorithms. Performing multiple iterations of training models with different feature subsets can be disadvantageous due to the computational expense.

To address the challenge of the computationally expensive process of evaluating feature subsets, this study explores the use of two neural network architectures as final predictive models. The key advantage is that these models do not require retraining when low-relevance features, as identified by feature selection algorithms, are removed. The first architecture is a multi-layer perceptron (MLP), where the weights of the removed features are pruned.

The second architecture is the FT-Transformer [7], which allows the model to generate predictions even when the embeddings of certain features are excluded from the input. Our findings show that these neural network models can perform competitively even without the need for retraining. When we removed up to 60% of low-relevance features, we observed performance degradations of less than 10%.

Crucially, by avoiding retraining, we were able to speed up the decision-making process by up to 300 times compared to the iterative training approach, especially for large datasets. Furthermore, we discovered that using up to 20% of the low-relevance features, the non-retrained neural networks could be employed as final predictive models without significant performance

**Table 2.** Properties of selected datasets. Datasets are sorted by their number of features

| Dataset | # Features | # Instances |
|---|---|---|
| volkert | 181 | 58310 |
| jasmine | 145 | 2984 |
| nomao | 119 | 34465 |
| kr-vs-kp | 37 | 3196 |
| sylvine | 21 | 5124 |
| australian | 15 | 690 |
| adult | 15 | 48842 |

loss compared to retraining-based methods. The main contributions of this study are as follows:

1. We introduce a straightforward pruning rule for MLPs when removing features.
2. We conduct a comparison of the proposed neural network architectures against retraining XGBoost, one of the most prominent models for tabular data. This comparison allowed us to determine the degradation rate and the evaluation speed when reducing the number of features across seven datasets.
3. We show that non-retrained neural networks remain competitive, achieving a degradation rate of less than 10% even when features are removed.
4. We demonstrate that pruned MLPs can be up to 300 times faster than tree-based methods when evaluating the impact of feature reduction.
5. We show that decision trees as feature selection algorithms generally achieve a small degradation rate.

The remainder of this paper is organized as follows: Section 2 provides background on the feature selection techniques and their considerations. In Section 3, we introduce the pruning rule for the MLPs. Additionally, we explain why the FT-Transformer has the ability to produce predictions even when features are removed.

Section 4 details the models selected for comparison and their optimization process. It also describes the datasets employed and their preprocessing. Finally, we outline the feature selection procedure used. Section 5 presents the performance degradations and execution times achieved by each approach, demonstrating the benefits of the proposed methods. Finally, Section 6 summarizes the key findings.

## 2 Background

Feature selection refers to the process of determining which features should be included in a model. From a practical standpoint, a model with fewer features can be more interpretable and less expensive to operate [14]. For example, a neural network with a lower number of features requires fewer parameters. Similarly, a tree-based model with fewer features requires fewer splits, resulting in shallower trees.

These model reductions lead to faster training and inference times, directly impacting the computational costs, whether in terms of time or monetary expense (e.g., when performed in a cloud environment).

Another perspective is that if a variable requires high-precision equipment to measure, and it is not as relevant as other features, we can discard its measurement, thereby saving costs. This highlights how feature selection can optimize the model complexity and the data collection process. The feature selection methods are classified into three types [17]:

1. **Filter Methods.** Rank the features based on their scores in various statistical tests for their correlation with the prediction target. The top-ranked features are kept, while the others are removed.

   Some key benefits of filter methods are that they are independent of the model used, are fast and scalable, capture feature dependencies, and reduce the overfitting risk.

   A simple approach is to rank the features based on their linear correlation with the target variable. The features with the highest correlation are placed at the top of the ranking.

2. **Wrapper Methods.** Identify the best performing set of features for a specific model by using the model's own performance as the metric to guide the selection of the optimal feature subset [8]. This is why wrapper methods typically outperform filtering methods regarding model performance, being its main advantage [12, 2, 6]. One example of a wrapper feature selection method is the F-Test.

   This approach creates a candidate linear model by removing a feature, and then statistically compares the similarity between the error means when using all features versus when one feature is removed, using an F-score. If the difference in error means is not statistically significant, the removed feature is considered non-relevant [22, 14].

3. **Embedded Methods.** Integrate the feature selection process directly into the model. During the training step, the model determines the relevance of each feature through its parameters or decision steps, to achieve the best evaluation score.

   Once the model is trained, the importance of each feature can be recovered and ranked to perform the feature selection. Embedded methods are faster than wrapper methods, as they only require a single training phase.

   One example of an embedded method is using a decision tree, where the feature importance is computed as the normalized total reduction in the Gini index brought about by that feature [23]. Another approach is to use the coefficients of a linear model, where a feature's importance is determined by its coefficient's magnitude, as long as the features are properly normalized [22].

Each feature selection method has its own benefits and drawbacks when compared to the others. As a result, no single algorithm can be considered universally optimal for all problems. This is why exploring and utilizing various feature selection methods is generally encouraged. However, it's important to carefully consider the trade-offs between the effectiveness and efficiency of each approach.

**Table 3.** Test balanced accuracy for each considered method when training in the whole set of features

| Dataset | FT-Transformer | MLP | XGBoost |
|---|---|---|---|
| volkert | 63.578 | 60.139 | 59.494 |
| jasmine | 80.477 | 77.255 | 80.560 |
| nomao | 94.813 | 94.010 | 96.217 |
| kr-vs-kp | 99.685 | 99.211 | 97.348 |
| sylvine | 94.123 | 92.955 | 94.689 |
| australian | 85.448 | 86.129 | 89.067 |
| adult | 79.187 | 78.774 | 76.816 |

## 3 Methodology

In this section, we outline the approaches taken to avoid retraining neural networks when relevant features are removed. We explain the pruning rule applied to a Multi-layer Perceptron (MLP). This approach allows us to efficiently update the model when removing less relevant features, without the need for full retraining.

Next, we briefly describe the FT-Transformer architecture and its ability to generate predictions even when the embeddings of certain features are excluded from the input.

This property of the FT-Transformer enables feature removals without any changes to the model structure or the need for retraining. The key benefit of these approaches is the ability to quickly assess the impact of removing low-relevance features, without the overhead of retraining the entire model for each feature subset evaluation.

### 3.1 Pruned Multi-Layer Perceptron

The Multi-layer Perceptron (MLP) is a fundamental approach for tabular data. It consists of a composition of affine transformations, including non-linearities, to create complex decision boundaries or simulate complex behaviors. The first layer of an MLP using a ReLU non-linearity can be expressed as:

$$\mathrm{ReLU}(\mathbf{W}\mathbf{x} + \mathbf{b}) \in \mathbb{R}^h, \tag{1}$$

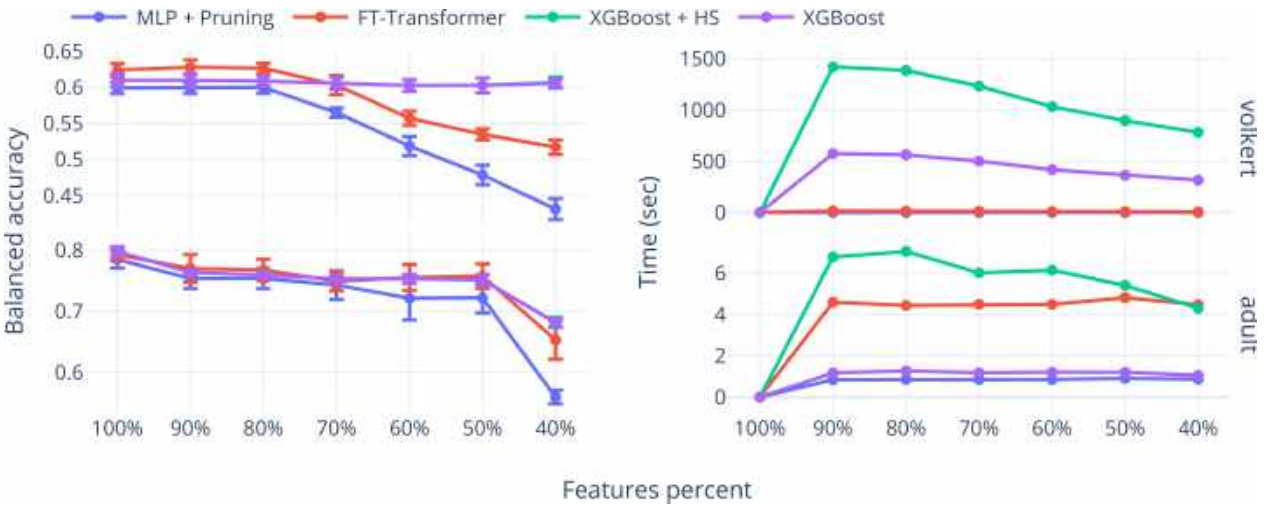



**Fig. 1.** Mean and standard deviation of cross-validation balanced accuracy for the best-performing architecture in each approach (left) and the elapsed time required to determine the balanced accuracies shown on the left (right), for the volkert and adult datasets

where $\mathbf{x} \in \mathbb{R}^m$ is the vector of input features, $\mathbf{W} \in \mathbb{R}^{h \times m}$ is a matrix of trainable weights, and $\mathbf{b} \in \mathbb{R}^h$ is a vector of trainable biases. When removing a feature, we can simply remove the weights associated with those features, and the output's dimension will not change, keeping the rest of the architecture invariant to the removals.

It is, for a subset of features $\mathcal{F} \subset \{1, \cdots, m\}$ we index the matrix of weights and the features vector, computing. When removing a feature, we can remove the weights associated with that feature from the weight matrix $\mathbf{W}$.

This keeps the output dimension $h$ unchanged, and the rest of the architecture remains invariant to the feature removals. For a subset of features $\mathcal{F} \subset \{1, \cdots, m\}$, we can index the weight matrix and feature vector accordingly:

$$\mathrm{ReLU}(\mathbf{W}_{\cdot,\mathcal{F}}\mathbf{x}_{\mathcal{F}} + \mathbf{b}) \in \mathbb{R}^h. \tag{2}$$

This pruning approach allows us to efficiently update the MLP model when removing less relevant features, without the need for full retraining.

### 3.2 FT-Transformer

The FT-Transformer [7] is an approach that combines a Feature Tokenizer with a Transformer architecture for tabular data. It creates embeddings for numerical and categorical variables (feature tokenization process) processed by a Transformer [19] variant. For numerical features, the embedding process involves applying an independent Multi-Layer Perceptron (MLP) to each feature.

For categorical features, the embedding process consists of applying an ordinal encoding to each feature and creating a look-up table for each one, similar to the approach used in Natural Language Processing tasks [19, 5]. Once tokenized, the embeddings are combined with an additional classification token [CLS] [5] at the beginning of the sequence.

This sequence is then processed by a Transformer encoder variant, which includes a PreNorm layer [21] and removes the first normalization from the first Transformer layer. Finally, the output embedding corresponding to the [CLS] token is passed through layer normalization [1], a ReLU activation function, and a single-layer perceptron with an identity activation function to make predictions.

ISSN 2007-9737



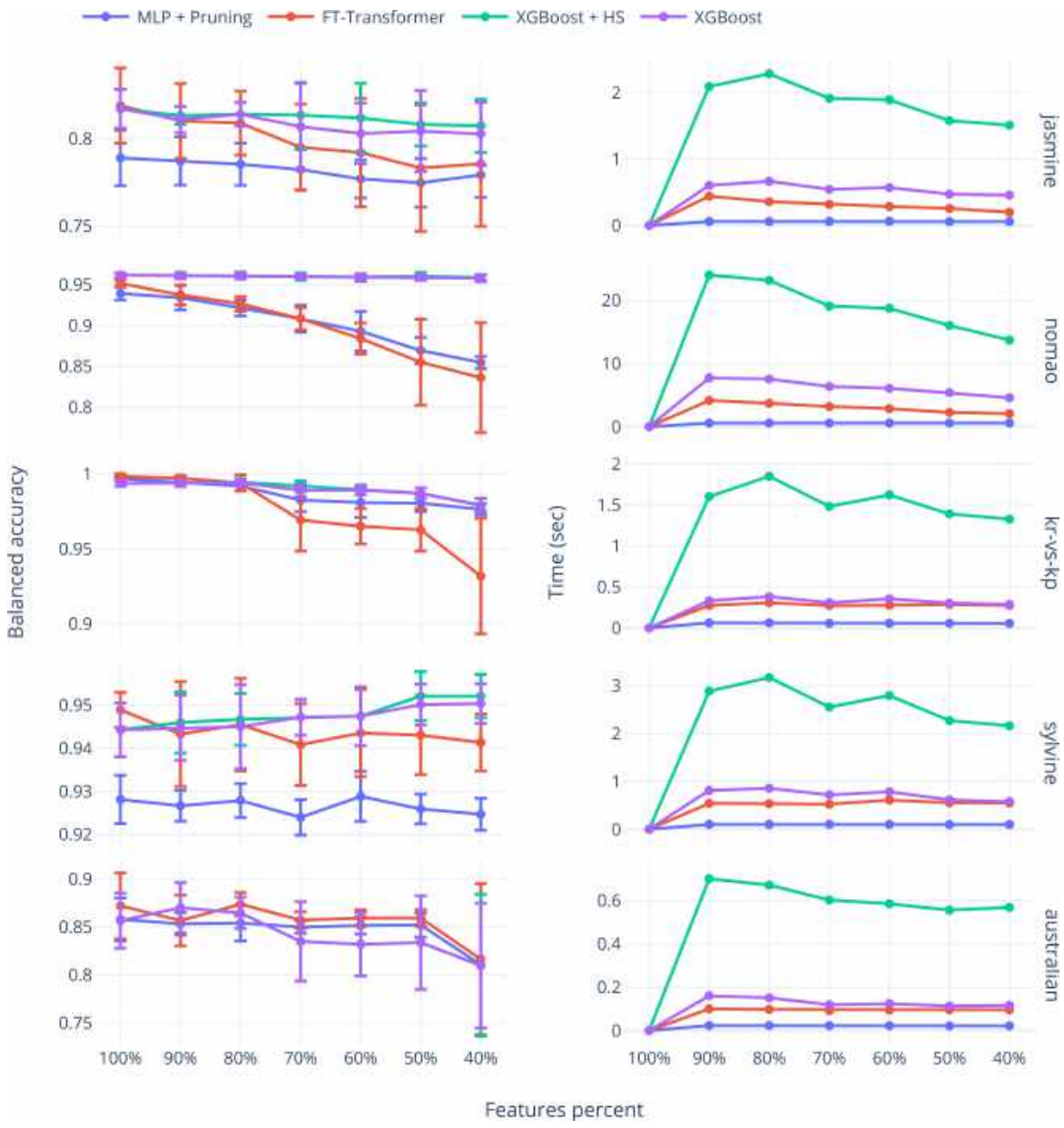



**Fig. 2.** Mean and standard deviation of cross-validation balanced accuracy for the best-performing architecture in each approach (left) and the elapsed time required to determine the balanced accuracies shown on the left (right), for the jasmine, nomao, kr-vs-kp, sylvine, and australian datasets

The FT-Transformer's capacity to generate predictions even when the embeddings of certain features are excluded from the input is enabled by its Multi-head Self-Attention (MHSA) mechanism as follows:

Let's consider the input to the FT-Transformer as a matrix of embedded features $\mathbf{E} \in \mathbb{R}^{(m+1)\times d}$, where $m+1$ represents the number of features plus the additional [CLS] token, and $d$ is the embedding dimension.

**Table 4.** Test balanced accuracy for each considered method when training using the 80% of the features selected using the decision-tree-based algorithm

| Dataset | FT-Transformer | MLP + Pruning | XGBoost | XGBoost + HS |
|---|---|---|---|---|
| volkert | 63.611 | 60.178 | 60.126 | 60.126 |
| jasmine | 80.606 | 77.602 | 80.848 | 80.848 |
| nomao | 94.189 | 94.639 | 96.320 | 96.215 |
| kr-vs-kp | 99.369 | 98.429 | 97.190 | 99.687 |
| sylvine | 93.931 | 93.051 | 95.370 | 95.466 |
| australian | 87.428 | 82.063 | 89.067 | 89.067 |
| adult | 75.731 | 74.390 | 73.676 | 73.676 |

The MHSA computation for a single layer of the FT-Transformer can be expressed as:

$$\text{Softmax}\left(\frac{(\mathbf{E}\mathbf{W}_Q)(\mathbf{E}\mathbf{W}_K)^T}{\sqrt{d}}\right)(\mathbf{E}\mathbf{W}_V) \in \mathbb{R}^{(m+1)\times d}, \quad (3)$$

where usually, $\mathbf{W}_Q \in \mathbb{R}^{d\times d}$, $\mathbf{W}_K \in \mathbb{R}^{d\times d}$, and $\mathbf{W}_V \in \mathbb{R}^{d\times d}$ are matrices of trainable parameters for the queries, keys, and values respectively. Whether we remove features from the input embedding matrix $\mathbf{E}$, the MHSA and subsequent operations can be computed in the same way, without affecting the overall architecture structure and then avoid the retraining procedure.

However, the output of the MHSA and the output of the Transformer encoder will now have a dimension of $(\mathcal{F}+1)\times d$, where $|\mathcal{F}|$ is the cardinality of the subset of retained features. In this study, we employed a modified version of the FT-Transformer. In contrast to the original architecture, we retained the first layer normalization step, and we did not apply layer normalization to the [CLS] token representation.

## 4 Experiments

This section outlines the experimental setup used to evaluate the impact of removing low-relevant features on non-retrained neural networks. We begin by describing the approaches taken for the model evaluation and optimization processes.

Next, we provide details on the datasets used in the study, as well as the data preprocessing steps applied. Finally, we outline the feature selection procedure that was followed, including the specific algorithms employed.

### 4.1 Compared Approaches

To compare the performance degradation of the Pruned MLP and FT-Transformer models when removing low-relevance features without retraining, we employed the XGBoost [4] algorithm as a baseline. The comparisons were made by removing features in increments of 10%, from 100% of the features down to 40%.

For the neural network models, we first performed a hyperparameter search using the full set of features, employing a grid search strategy. We then kept the architecture with the highest balanced accuracy fixed, and applied the pruning rules described in Section 3 for each feature removal scenario.

For the XGBoost models we took two approaches. The first (XGBoost) perform an initial grid search using the full feature set, then keep the highest performing architecture fixed for each feature removal, retraining the model.

The second approach (XGBoost + HS) performs the hyperparameter search for each subset of features tested. All models were optimized using the cross-entropy loss. The neural networks used the AdamW optimizer for 150

**Table 5.** Mean degradation percentages for each dataset, considering features percentages ranging from 100% to 40%

| Dataset | FT-Transformer | MLP | XGBoost + HS | XGBoost |
|---|---|---|---|---|
| volkert | $6.330 \pm 7.541$ | $9.604 \pm 11.250$ | $0.525 \pm 0.499$ | $0.553 \pm 0.504$ |
| jasmine | $2.418 \pm 1.660$ | $0.872 \pm 0.681$ | $0.608 \pm 0.416$ | $1.065 \pm 0.695$ |
| nomao | $5.438 \pm 4.538$ | $3.865 \pm 3.422$ | $0.125 \pm 0.131$ | $0.206 \pm 0.139$ |
| kr-vs-kp | $2.439 \pm 2.432$ | $1.065 \pm 0.804$ | $0.376 \pm 0.547$ | $0.436 \pm 0.544$ |
| sylvine | $0.540 \pm 0.287$ | $0.166 \pm 0.197$ | $-0.392 \pm 0.318$ | $-0.292 \pm 0.266$ |
| australian | $1.790 \pm 2.168$ | $1.259 \pm 1.887$ | $1.570 \pm 2.493$ | $1.576 \pm 2.504$ |
| adult | $5.426 \pm 5.646$ | $8.303 \pm 9.424$ | $5.820 \pm 4.348$ | $5.834 \pm 4.383$ |

epochs with 30 early stopping patience steps. The XGBoost models were trained over 150 estimators with 30 early stopping patience steps. The hyperparameter search spaces for each model are described in Table 1.

### 4.2 Datasets and Preprocessing

We selected seven datasets from the OpenML repository [3]. The selected datasets were: volkert, jasmine, nomao, kr-vs-kp, sylvine, australian, and adult. The properties of these datasets are summarized in Table 2.

The datasets were split into 80% for training and 20% for testing. The training partition was further divided into five stratified folds to perform cross-validation (CV). Numerical features were normalized to have a mean of 0 and a standard deviation of 1. Missing values were imputed using the KNNImputer from scikit-learn [16] with $k = 10$ nearest neighbors.

For the FT-Transformer and XGBoost models, categorical features were encoded using an ordinal encoder, where the zero code was reserved for null values. In contrast, the MLP model used one-hot encoding for the categorical features.

### 4.3 Feature Selection

We considered three approaches for feature selection: the F-test, the decision tree, and linear model embedded methods. While embedded methods are particularly well-suited when applying feature reduction directly to the model that embeds the feature relevance, their feature ranking can still be leveraged and applied to other models as well. We used a specific methodology to perform the feature selection. First, we standardized the numerical features to have a mean of 0 and a standard deviation of 1, and one-hot encoded the categorical features. All missing values were replaced with zeros. Next, we performed feature selection using each of the three algorithms.

When a feature was selected among the top $k$ features by a given algorithm, we marked the corresponding original feature (before one-hot encoding) as one of the top $k$ features.

This selection process continued until $k$ features from the original dataset were marked. In the cross-validation setup, the feature selection was performed independently for each fold. For the final testing, the feature selection was done using the training partition.

## 5 Results

### 5.1 General Performance

We verify that every proposed method achieves similar results for every dataset when training using the entire set of features. This is because if the proposed methods were not competitive in the first instance, they would be discarded as an initial model. Table 3 shows the balanced accuracy in the test set for the fixed architectures from the cross-validation scores. Due to the selection methodology, the XGboost and XGBoost + HS models are the same when training using all characteristics.

**Table 6.** Speed ratios for each approach and dataset, considering feature percentages from 100% to 40%. The MLP was omitted since it was the base for the comparison

| Dataset | FT-Transformer | XGBoost + HS | XGBoost |
|---|---|---|---|
| volkert | $\times 7.6 - \times 18.2$ | $\times 905.6 - \times 1533.5$ | $\times 366.8 - \times 621.4$ |
| jasmine | $\times 3.4 - \times 7.3$ | $\times 25.5 - \times 36.8$ | $\times 7.7 - \times 10.7$ |
| nomao | $\times 3.5 - \times 7.2$ | $\times 23.0 - \times 41.3$ | $\times 7.7 - \times 13.3$ |
| kr-vs-kp | $\times 4.4 - \times 5.0$ | $\times 23.8 - \times 29.5$ | $\times 5.2 - \times 6.1$ |
| sylvine | $\times 5.0 - \times 6.1$ | $\times 21.6 - \times 30.8$ | $\times 5.8 - \times 8.3$ |
| australian | $\times 4.1 - \times 4.5$ | $\times 26.0 - \times 30.6$ | $\times 5.3 - \times 7.0$ |
| adult | $\times 5.1 - \times 5.4$ | $\times 4.9 - \times 8.2$ | $\times 1.2 - \times 1.5$ |

### 5.2 Performance and Efficency

The removal of low-relevant features can affect the models in two ways. First, the information reduction may impact the models' performance. Second, the models may become more efficient, as they require fewer parameters and operations.

To evaluate these trade-offs, we assess the benefits and disadvantages of using non-retrained networks in both aspects. Figure 1 presents two cases of study: the volkert and adult datasets, which have the highest and lowest number of features, respectively.

The plots include the mean and standard deviations of the cross-validation balanced accuracy against the time required to obtain the balanced accuracies for each evaluation method when using the decision-tree-based feature selection algorithm.

As will be shown in Section 5.3, the decision-tree-based algorithm was the best method for reducing the number of features. The volkert dataset demonstrates a case where the proposed methods excel in efficiency, requiring significantly less time compared to methods that require retraining. However, the effectiveness is only maintained when the removal of low-relevance features is up to 20%.

Conversely, in the adult dataset, the required time for the FT-Transformer, even without retraining, is higher than the time required for XGBoost. Regarding the balanced accuracy degradation, both methods achieve similar results.

The adult dataset represents a poor case for the application of the proposed methods. Figure 2 presents the cross-validation balanced accuracies and the required time for the other datasets. Across all datasets, the time needed to evaluate the low-relevance feature removals is lower than the time required for methods that need retraining.

Notably, for every dataset, the degradation in the balanced accuracy of at least one of the proposed methods is very similar to the methods requiring retraining, when the number of selected features ranges between 100% and 80%.

Table 4 shows the balanced accuracies in the test set when using 80% of the features. These results are comparable to those presented in Table 3. The key observation is that for every dataset, the results of every method remain competitive, and even for the volkert, jasmine, sylvine, and australian datasets, the balanced accuracy is higher when using a lower number of features. This highlights the importance of performing feature selection for removing redundant or noisy variables.

To quantify and summarize the observed performance results, we computed the degradation percentage with respect to the balanced accuracy achieved when using 100% of the features for a specific dataset and every method. Given a specific dataset $\mathcal{D}$, a subset of features $\mathcal{F}$, and a method's space $\mathcal{H}$ (e.g., all considered architectures in the XGBoost + HS space), the degradation percentage $d$ is calculated as:

ISSN 2007-9737



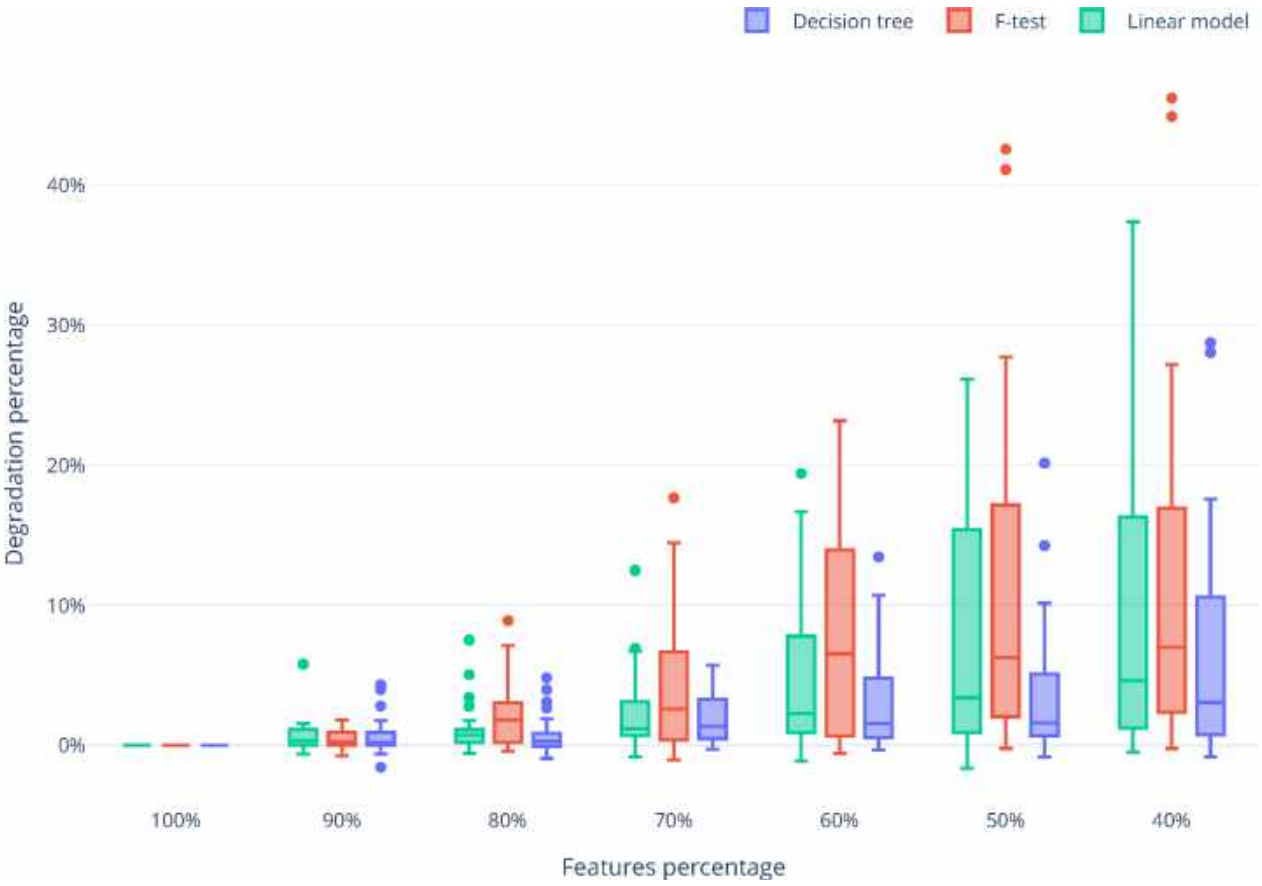



**Fig. 3.** Percentage of features selected versus the degradation ratio for each algorithm used for feature selection

$$d(\mathcal{D}, \mathcal{H}, \mathcal{F}) = \left[1 - \frac{\max\limits_{h \in \mathcal{H}} \mathcal{B}(\mathcal{D}_{\mathcal{F}}, h)}{\max\limits_{h \in \mathcal{H}} \mathcal{B}(\mathcal{D}, h)}\right] \cdot 100, \qquad (4)$$

where $\mathcal{D}_{\mathcal{F}}$ is the dataset containing only the selected features in $\mathcal{F}$ and $\mathcal{B}$ is the mean cross-validation balanced accuracy.

Table 5 shows the mean and standard deviation of the degradation percentages, averaged across all subsets of features generated, ranging from 100% to 40% for each dataset.

Negative degradations indicate an improvement over the models trained with the entire feature set. As expected, the highest degradation percentages correspond to methods that were not retrained.

However, the mean degradation percentage for the proposed methods remains below 10%. In contrast to the degradation, the models that do not require retraining exhibit significantly shorter required times.

To compare and summarize the required time for each method, we considered the ratio between every method and the fastest method, which is the MLP approach. Let $\mathcal{D}_{\mathcal{F}}$ the dataset containing only the features in $\mathcal{F}$, $\mathcal{H}$ the compared method's space, $\mathcal{H}_{MLP}$ the MLP space, and $\mathcal{T}$ the time required to complete the experiment for a given dataset and a given method, the speed ratio $s$ is calculated as:

$$s(\mathcal{D}_{\mathcal{F}}, \mathcal{H}) = \frac{\mathcal{T}(\mathcal{D}_{\mathcal{F}}, \mathcal{H})}{\mathcal{T}(\mathcal{D}_{\mathcal{F}}, \mathcal{H}_{MLP})}. \qquad (5)$$

Table 6 presents the minimum and maximum speed ratios of all subsets of features generated, ranging from 100% to 40% for each dataset. As observed, the FT-Transformer is generally the fastest method, excluding the MLP approach, while the XGBoost + HS is the slowest. Even when the XGBoost + HS is the correct way to perform the tests, the XGBoost was a heuristic method that could work but invested less time.

In this case, the proposed methods outperform the efficiency, showing to be up to 300 times faster. It is important to note that the speed ratios of these approaches are closer when using small datasets. However, as the number of instances and features increases, the difference in speed ratios becomes more significant.

### 5.3 Feature Selection Algorithms Performance

To determine the best feature selection algorithm, we analyze the distributions of the degradation percentages for each feature selection algorithm. Figure 3 shows the distribution of the degradation ratio, computed as in Equation 4, for each feature selection algorithm and percentage of features selected, across all datasets.

As observed, the decision tree algorithm generally shows the lowest degradation median and the smallest interquartile range. This indicates that the concentration of the degradations tends to be the smallest among all feature selection algorithms considered, making it the most suitable option for feature removal.

## 6 Conclusions

In this study, we have evaluated the impact of removing low-relevance features using non-retrained networks. To select low-relevant features, we employed three feature selection algorithms: The embedded decision tree, the F-test, and the embedded linear model. To avoid the retraining procedure, we proposed the use of two architectures: a pruned MLP and the FT-Transformer, leveraging their abilities to create predictions even when certain features are removed.

Our results demonstrate the viability of this approach. When removing up to 60% of low-relevance features, we achieved performance degradations lower than 10%. Crucially, by avoiding retraining, we were able to speed up the decision-making process by up to 300 times, especially for large datasets.

Moreover, we found that using up to 20% of the low-relevant features, the non-retrained neural networks could be employed as final predictive models without sacrificing significant performance compared to retraining-based methods. This last finding is particularly noteworthy.

When working with large datasets, the high costs associated with retraining models as features are removed can be prohibitive. However, by leveraging our proposed non-retrained techniques, these expenses can be largely avoided.

Furthermore, our analysis of the degradation distributions revealed that the embedded decision-tree-based feature selection algorithm outperformed both linear model coefficients and the F-test in identifying the most appropriate low-relevant features to remove.

## Acknowledgments

Uriel Corona-Bermúdez and Erendira Corona-Bermúdez thanks CONAHCyT for the scholarship granted towards pursuing his graduate studies.

## References

1. **Ba, J. L., Kiros, J. R., Hinton, G. E. (2016).** Layer normalization. arXiv. DOI: 10.48550/arxiv.1607.06450.
2. **Bee-Wah, Y., Ibrahim, N., Abdul-Hamid, H., Abdul-Rahman, S., Simon, F. (2018).** Feature selection methods: Case of filter and wrapper approaches for maximising classification accuracy. Pertanika Journal of Science and Technology, Vol. 26, No. 1, pp. 329–340.

3. **Casalicchio, G., Bossek, J., Lang, M., Kirchhoff, D., Kerschke, P., Hofner, B., Seibold, H., Vanschoren, J., Bischl, B. (2017).** OpenML: An R package to connect to the machine learning platform OpenML. Computational Statistics, Vol. 34, No. 3, pp. 977–991. DOI: 10.1007/s00180-017-0742-2.

4. **Chen, T., Guestrin, C. (2016).** XGBoost: A scalable tree boosting system. Proceedings of the 22nd Association for Computing Machinery's Special Interest Group on Knowledge Discovery and Data Mining International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, pp. 785–794. DOI: 10.1145/2939672.2939785.

5. **Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018).** BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 4171–4186. DOI: 10.18653/v1/n19-1423.

6. **Ghosh, M., Guha, R., Sarkar, R., Abraham, A. (2019).** A wrapper-filter feature selection technique based on ant colony optimization. Neural Computing and Applications, Vol. 32, No. 12, pp. 7839–7857. DOI: 10.1007/s00521-019-04171-3.

7. **Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A. (2021).** Revisiting deep learning models for tabular data. 35th Conference on Neural Information Processing System, pp. 1–25. DOI: 10.48550/arXiv.2106.11959.

8. **Guyon, I., Elisseeff, A. (2003).** An introduction of variable and feature selection. The Journal of Machine Learning Research, Vol. 3, pp. 1157–1182.

9. **Ho, T. K. (1995).** Random decision forests. Proceedings of 3rd International Conference on Document Analysis and Recognition, Vol. 1, pp. 278–282. DOI: 10.1109/ICDAR.1995.598994.

10. **Hollmann, N., Müller, S., Eggensperger, K., Hutter, F. (2022).** TabPFN: A transformer that solves small tabular classification problems in a second. International Conference on Learning Representations, pp. 1–37. DOI: 10.48550/arXiv.2207.01848.

11. **Huang, X., Khetan, A., Cvitkovic, M., Karnin, Z. (2020).** TabTransformer: Tabular data modeling using contextual embeddings. arXiv. DOI: 10.48550/ARXIV.2012.06678.

12. **Inza, I., Larrañaga, P., Blanco, R., Cerrolaza, A. J. (2004).** Filter versus wrapper gene selection approaches in DNA microarray domains. Artificial Intelligence in Medicine, Vol. 31, No. 2, pp. 91–103. DOI: 10.1016/j.artmed.2004.01.007.

13. **Kadra, A., Lindauer, M., Hutter, F., Grabocka, J. (2021).** Well-tuned simple nets excel on tabular datasets. 35th Conference on Neural Information Processing Systems, pp. 1–14.

14. **Kuhn, M., Johnson, K. (2013).** Applied Predictive Modeling. Springer New York. DOI: 10.1007/978-1-4614-6849-3.

15. **Morey, T. (2015).** Customer data: Designing for transparency and trust. Harvard Business Review, Vol. May 2015, pp. 96–105.

16. **Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2011).** Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, Vol. 12, No. 85, pp. 2825–2830.

17. **Pudjihartono, N., Fadason, T., Kempa-Liehr, A. W., O'Sullivan, J. M. (2022).** A review of feature selection methods for machine learning-based disease risk prediction. Frontiers in Bioinformatics, Vol. 2. DOI: 10.3389/fbinf.2022.927312.

18. **Russell, S., Norvig, P. (2010).** Artificial Intelligence: A Modern Approach. Prentice Hall.

19. **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017).** Attention is all you need. Advances in Neural Information Processing Systems, Vol. 30.

20. **Venkat, N. (2018).** The curse of dimensionality: Inside out. DOI: 10.13140/RG.2.2.29631.36006.

21. **Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., Chao, L. S. (2019).** Learning deep transformer models for machine translation. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1810–1822. DOI: 10.48550/ARXIV.1906.01787.

22. **Wooldridge, J. M. (2013).** Introductory econometrics: A modern approach. South-Western.

23. **Zhang, C., Ma, Y. (2012).** Ensemble machine learning: Methods and applications. Springer Publishing Company, Incorporated.

*Article received on 16/04/2024; accepted on 28/05/2024.*
**Corresponding author is Ricardo Menchaca-Méndez.*

ISSN 2007-9737

# Respiratory Disease Pre-Diagnosis through a Novel Pattern Classification Algorithm based on Associative Memories

Osvaldo D. Velazquez-Gonzalez[1], Yenny Villuendas-Rey[2,*]

[1] Instituto Politécnico Nacional,
Centro de Investigación en Computación
Mexico

[2] Instituto Politécnico Nacional,
Centro de Innovación y Desarrollo Tecnológico en Cómputo,
Mexico

velazquezg2020@cic.ipn.mx, yvilluendasr@ipn.mx

**Abstract.** In this paper, the Subtractive Threshold Associative Classifier (STAC), a novel supervised machine learning model, is presented. The main contribution of the proposed model is to have the capability to adequately deal with medical dataset for the pre-diagnosis of respiratory disease and class imbalance data complexity without applying any other pre-processing technique, obtained competitive results. Furthermore, the proposed algorithm is interpretable and transparent, since the reasons why a test pattern was classified as belonging to a specific class. The experimental results were validated with the purpose of finding possible significant differences in performance; For this, statistical tests were used. It is necessary to emphasize that the experimental tests carried out allow us to verify that the novel proposed algorithm is competitive against the most used algorithms in the state of the art.

**Keywords.** Machine learning, pattern classification, associative memories, respiratory diseases.

## 1 Introduction

Early detection of diseases has been of utmost importance in recent years, due to the different benefits that can impact society, such as increasing the chances of survival in patients suffering from potentially fatal diseases [1]. Currently, the research carried out in the pre-diagnosis of diseases is notably relevant, specifically with great interest in minimizing errors in the early detection of lung diseases; this, due to the different benefits, such as increasing survival in patients, achieving a better recovery thanks to detection in a premature phase of the disease, implementing better clinical management of the patient, adopting public health and controlling possible outbreaks [1].

Recently, machine learning techniques applied to the field of medicine have become an increasingly important area of research at a global level, promoting the frequent emergence in the literature of works related to the development of novel and advanced models specialized in the pre-diagnosis of diseases, which makes it an active research topic [1].

A very important aspect related to medical pre-diagnosis of diseases is that most datasets related to this type of problems are imbalanced, which is not favorable to the pre-diagnosis of diseases using machine learning algorithms [2]. According to the National Cancer Institute of the US National Institutes of Health [3] respiratory diseases or lung diseases are pathological conditions that affect the lungs and other parts of the respiratory system.

There are two types of respiratory diseases [4]: infectious and chronic, which range from mild symptoms, such as the common cold, flu and pharyngitis, to life-threatening diseases such as pneumonia, pulmonary embolism, tuberculosis, asthma, lung cancer, pulmonary fibrosis, chronic obstructive pulmonary disease (COPD) and severe acute respiratory syndromes, such as COVID-19 disease [5].

According to the Forum of International Respiratory Societies (FIRS) and data from the World Health Organization (WHO), respiratory diseases are among the most important causes of death and disability worldwide [4, 6]. in 2019, respiratory diseases were three of the top ten causes of death, causing more than 8 million deaths annually [6]. COVID-19 pandemic that began in 2020 has affected around 400 million people until 2022, claiming the lives of more than 6 million people worldwide [7], and in Mexico has left more than 320 thousand dead in a period of two years [7], making it the main cause of death nationwide during the first half of the year. of the year 2021 [8].

On the other hand, the diagnosis of respiratory diseases is usually made using different methods, both invasive and non-invasive; for example, one of the most common is through computer-aided diagnosis (CAD). Some of the most frequent techniques used within CAD to diagnose respiratory diseases are: chest x-ray, computed tomography and magnetic resonance [9].

The diagnostic methods presented in the literature have disadvantages and limitations, such as: special equipment, highly trained personnel, financing, and specialized studies, causing negative results when implementing these techniques in the diagnosis of diseases. Therefore, it is necessary to continue researching new methods or technologies that help make a better early diagnosis [10].

This is why machine learning techniques applied to medicine have become an increasingly important area of research around the world, as well as the application and development of novel models for the pre-diagnosis of diseases, which is a relevant research field [2, 11, 10].

On the other hand, the No Free Lunch theorem [12] proves and establishes that there is no classifier that is the best on any kind of dataset.

Given that associative models have been shown to be effective and efficient in achieving this minimization of errors, in the present research work a novel specialized classification machine learning model is proposed for the pre-diagnosis of respiratory diseases, called Subtractive Threshold Associative Classifier (STAC). The experimental tests carried out with the STAC allow us to verify that the new model is competitive in the state of the art.

The paper is organized in the following manner. Section 2 presents the related works, where a brief description of the different works published related to pre-diagnosis of respiratory diseases using machine learning is given. On the other hand, section 3 presents a brief description of the different datasets and algorithms used. In the section 4 some materials and methods are presented, highlighting the theoretical concepts that will support this work.

Likewise, the proposal of this work is addressed in section 5, and section 6 shows the results achieved from the experimental phase to evaluate the viability of the new model; Finally, conclusions and proposals for future work are presented in section 7.

## 2 Related Works

Recently, the researches focused on the pre-diagnosis of respiratory diseases has gained momentum worldwide, with broad interest in improving the early detection of respiratory diseases. Currently, to make this type of diagnosis in respiratory diseases, different methods are applied, both invasive and non-invasive. Some of the most common methods are computer-aided diagnosis (CAD), pulmonary function tests (such as spirometry, lung volume, gas diffusion, and bronchoscopy), microbiological diagnoses, and molecular biology-based diagnoses [10, 13].

Within the state of the art, several works have addressed the topic of pre-diagnosis of respiratory diseases applying Machine Learning techniques.

Maleki et al. [14] addressed the pre-diagnosis of lung cancer, one of the most common diseases among humans worldwide. For the classification task, the authors use general data referring to patients suffering from lung cancer. The dataset under consideration for the study of this research is made up of 100 patterns with 23 features, which describe information about the patients. Finally, in the classification process, in this case used to diagnose lung cancer, the kNN algorithm is applied, which the model reaches an accuracy equal to 1.

On the other hand, Spathis et al. [15] studied the prevention, diagnosis and early detection of respiratory diseases, such as asthma and chronic obstructive pulmonary disease (COPD). The authors carried out a comparative study applying different algorithms, such as: Naïve Bayes, logistic regression, multilayer perceptron neural networks (MLP), support vector machines (SVM), near neighbors (kNN), decision trees, and Random Forest. As a result of the comparative analysis, it was observed that the best classification algorithm for diagnosing asthma and COPD is the random forest algorithm, which obtained the highest accuracy values.

In the work presented by Cardoso et al. [16] proposed a new methodology to diagnose interstitial lung disease (ILD) obtained better results in diagnosis over the related works of this art. The authors applied feature extraction techniques to reduce dimensionality, such as Principal Component Analysis (PCA) and linear discriminant analysis applying models as SVM, kNN, and feedforward deep neural network, which reached the best performances.

Finkelstein et al. [17] used three machine learning algorithms (Naïve Bayes, adaptive Bayesian network, and support vector machines) to perform a comparative analysis on the early detection of exacerbations in adult patients with asthma. The models reached excellent performance at the metrics sensitivity and specificity.

Amaral et al. [18] developed a medical decision support system to simplify clinical use as well as improve the diagnosis of airway obstruction in patients suffering from asthma. The comparative study used the principal component analysis (PCA) technique to try to improve classification performance.

However, based on the results obtained, it was concluded that the use of dimensionality reduction does not significantly benefit the performance of the algorithms in this particular case. It is shown that the best algorithm to diagnose airway obstruction in patients with asthma is the kNN algorithm with a value of k=1 and the AdaBoost classifier, which allow sick patients to be classified with outstanding performance.

With the aim of increasing the survival rate in patients suffering from lung cancer, Radhika et al. [19] propose to diagnose lung cancer early in affected patients using: Naïve Bayes, Support Vector Machines (SVM) and logistic regression.

In another trend, novel machine learning techniques have recently emerged, which work adequately using images as input information, easily outperforming other algorithms in this type of tasks [20]. These techniques are called deep learning (Deep Learning) or convolutional neural networks (CNN).

For example, Xiong et al. [21] proposed a specialized CNN model to recognize Mycobacterium tuberculosis using tissue samples treated with acid-fast staining, where after the experiments carried out, the new proposed CNN model achieved sensitivity values of 97.94% and specificity of 83.65%.

Another example under the same group of algorithms is Christe et al. [22] presenting a study to evaluate the performance of a new computer-aided diagnosis system based on a convolutional neural network (CNN) for automatic classification of high-resolution computed tomography images into four radiological diagnostic categories. Likewise, there are related works where techniques related to deep learning are applied to pre-diagnose patients suffering from respiratory diseases of COVID-19 or pneumonia [11].

Finally, within the related literature there are works where pulmonary acoustic signals from patients' thoracic ultrasound have been used, in order to make diagnoses of diseases linked to the chest, such as pleural effusion, atelectasis, pneumothorax and pneumonia [23]. For example, Pham et al. [24] make use of convolutional neural networks to detect respiratory diseases from recordings of respiratory sounds, using traditional machine learning models, such as Support Vector Machines (SVM) and Nearest Neighbors (kNN) algorithms.

## 3 Datasets and Algorithms

In this section, a brief description of the pattern classification algorithm applied in the present work and the used datasets related to respiratory pre-diagnosis diseases are presented.

**Table 1.** Description of the selected datasets

| Datasets | Features | | Patterns | IR | Classes |
|---|---|---|---|---|---|
| | Categorical | Numerical | | | |
| Post-operative | 8 | 0 | 90 | 32.00 | 3 |
| Thyroid | 0 | 21 | 7200 | 40.10 | 3 |
| Newt-thyroid1 | 5 | 0 | 215 | 5.14 | 2 |
| Newt-thyroid2 | 5 | 0 | 215 | 5.14 | 2 |
| Thoracic-Surgery | 13 | 3 | 470 | 5.70 | 2 |
| Lung-Cancer | 0 | 52 | 32 | 1.40 | 3 |
| Survey Lung-Cancer | 14 | 1 | 309 | 6.90 | 2 |
| ACPs Lung Cancer | 38 | 0 | 901 | 31.25 | 4 |
| Exasens | 0 | 7 | 80 | 1.00 | 2 |
| Lymphography | 3 | 15 | 148 | 40.50 | 4 |
| Lymphography-NF | 3 | 15 | 148 | 23.60 | 2 |
| Primary-tumor | 16 | 1 | 336 | 42.00 | 18 |

### 3.1 Datasets Related to Respiratory Diseases

For this work, 12 datasets were selected in three different repositories, the Knowledge Extraction base on Evolutionary Learning (KEEL) repository [25] located at https://sci2s.ugr.es/keel/datasets.php, the Machine Learning Repository from the University of California at Irvine (UCI) [26] located at https://archive.ics.uci.edu/ml/index.php, and finally, the Kaggle repository located at https://www.kaggle.com/datasets. Of the 12, 10 datasets have an imbalance ratio (IR) greater than 1.5, which means have an imbalanced complexity. The IR ratio is calculated as the expression 1.

Detailed information about each of the selected datasets is shown in Table 1:

$$IR = \frac{Number\ of\ majority\ class\ patterns}{Number\ of\ minority\ class\ patterns}. \quad (1)$$

On the other hand, the 12 datasets mentioned above were selected because they include information on the most common respiratory diseases [5], such as pneumonia, pulmonary embolism, tuberculosis, asthma, lung cancer, pulmonary fibrosis, chronic obstructive pulmonary disease. (COPD) and severe acute respiratory syndromes.

Post-operative: This dataset comes from a study to determine where a patient should be sent after post-operative recovery, because hypothermia is a major risk post-surgery.

Thyroid: The task of classifying this dataset is to determine whether a given patient is healthy (normal) or suffers from hypothyroidism or hyperthyroidism.

Newt-thyroid1 and Newt-thyroid2: Both datasets represent an imbalanced version of the original Thyroid dataset. In the Newt-thyoid1 set, the positive class belongs to the hyperthyroidism class, and the patterns of the negative class are made up of the patterns of the rest of the classes.

Thoracic-Surgery: This dataset represents patients who underwent major lung resections for primary lung cancer between 2007 and 2009 at the Thoracic Surgery Center in Wroclaw.

Lung-Cancer: This dataset describes three types of pathological lung cancers. The objective of the data set is to classify these three types of cancers.

Survey Lung-Cancer: The classification task of this dataset is to detect whether or not a given

patient suffers from lung cancer, based on different variables collected from a survey.

ACPs Lung Cancer: This dataset represents information on peptides (amino acid code) and anticancer activity in lung cancer cell lines.

Exasens-COPD: This dataset aims (based on demographic information from saliva) to classify patients into four classes according to their membership: chronic obstructive pulmonary disease, COPD or COPD, asthma, respiratory infections and completely healthy patients.

Lymphography: The classification task of this dataset is to detect the presence of lymphomas in addition to their current status.

Lymphography-NF: This dataset is a two-class only version from the KEEL repository of the original Lymphography dataset. In this set, the positive class is made up of the “normal” and “fibrosis” classes while the negative class is made up of the rest of the classes.

Primary-tumor: This dataset aims to classify patients within 21 different classes, according to the type of tumor they suffer from.

### 3.2 Classification Algorithms

This section describes the pattern classification algorithms proposed to carry out the comparative study against the novel model presented in this work, which are applied to the datasets described in section 3.1. The algorithms presented below were selected because they comprise the most relevant models in the results table within the state of the art on topics related to pattern classification, as can be seen in [27, 28, 29].

Naîve Bayes [30] is a type of algorithm that belongs to probability-based classifiers. This classification algorithm is based on Bayes' Theorem, specifically considering all independent attributes from a probabilistic approach.

Another classifier used was the kNN or K-nearest neighbor algorithm [14], specifically the 1NN and 3NN models. In WEKA, the classifier algorithm is called Instance-Based (IBk).

Multilayer perceptron (MLP) [31] is a well-known classification algorithm within the literature on topics related to Machine Learning. MLP is a network composed of artificial neurons (also called units) interconnected with each other, forming three different types of layers, which are: the input layer, the hidden layer and finally the output layer (output layer).

Sequential minimal optimization (SMO) [32] is one of the most important and widely used optimization algorithms for support vector machines (SVM) within the state of the art when comparing classifiers. This classifier uses the sequential minimal optimization algorithm created by John Platt to train support vector machines using kernel functions based on linear, polynomial, radial basis or sigmoid functions.

And finally, the classifier C4.5 [33] is a decision tree, which is an extension of the ID3 algorithm. This type of classifier is highly recognized within the state of the art because it is explainable, it is based on information theory and its hierarchical structure allows us to see how the patterns of a data set are classified.

These algorithms were executed in the WEKA software in version 3.8, using the default parameters offered by the software.

## 4 Associative Memories

This section includes fundamental concepts of two pioneering models of associative memories, Steinbuch's Lernmatrix [34] and Willshaw's Correlograph [34], due to these models are the basis for the proposed model presented in section 5.

An associative memory $\boldsymbol{M}$ is a pattern input and output system (see equation 2), whose main objective is to learn to correctly recover complete patterns from input patterns, which can be altered with different types of noise (additive, subtractive or mixed) [34]:

$$x \rightarrow \boxed{\boldsymbol{M}} \rightarrow y. \tag{2}$$

There are two types of associative memories. Autoassociative memory, which meets the following conditions: $x^{\mu} = y^{\mu}\ \forall \mu \in \{1,2,\dots,p\}$. On the other hand, the memory is declared to be heteroassociative if it holds that $x^{\mu} \neq y^{\mu}\ \exists \mu \in \{1,2,\dots,p\}$ [15].

Associative memories are made up of two essential phases [15].

Learning phase. It consists of creating the associative memory (matrix) **M** that manages to store the *p* associations of the fundamental set.

Recovery phase. It consists of operating the associative memory (matrix) **M** with the objective of finding the sufficient conditions to obtain the fundamental output pattern $y^{\mu}$ from the fundamental input pattern $x^{\mu}$.

### 4.1 Steinbuch's Lernmatrix

The Steinbuch Lernmatrix is a heteroassociative memory, which can function equally as a binary pattern classification algorithm if the output patterns corresponding to each input pattern are correctly chosen.

#### 4.1.1 Learning Phase

The learning phase consists of finding a way to generate a matrix $\boldsymbol{M}$ that stores the information of the $p$ associations of the fundamental set. The process to determine each of the components $m_{ij}$ can be described in two steps [34].

1. Each of the components $m_{ij}$ of the matrix $\boldsymbol{M}$ is initialized to zeros.
2. Each component $m_{ij}$ is updated according to the rule $m_{ij} + \Delta m_{ij}$, where:

$$\Delta m_{ij} = \begin{cases} +\varepsilon & \text{if } y_i^{\mu} = 1 = x_j^{\mu}, \\ -\varepsilon & \text{if } y_i^{\mu} = 1 \text{ and } x_j^{\mu} = 0, \\ 0 & \text{In any other case,} \end{cases} \quad (3)$$

where each $\varepsilon$ represents any previously selected positive constant.

#### 4.1.2 Recovery Phase

The recovery or classification phase if used as a classifier consists of multiplying the previously trained memory $\boldsymbol{M}$ with a given unknown input vector, with the objective of finding the class to which the input vector belongs.

To carry out the recovery phase, it is necessary to calculate the $i$-*th* coordinate of the output vector (vector that represents the pattern class), which is obtained using the following expression [1, 34]:

$$y_i^{\omega} = \begin{cases} 1 \;\; if \;\; \sum_{j=1}^{n} m_{ij} \cdot x_j^{\omega} = \bigvee_{h=1}^{p} \left[ \sum_{j=1}^{n} m_{hj} \cdot x_j^{\omega} \right], \\ 0 \qquad\qquad \text{In any other case.} \end{cases} \quad (4)$$

### 4.2 Willshaw's Correlograph

The Willshaw's correlograph is an optical device, which can function as an associative memory. This associative memory works in the following way.

#### 4.2.1 Learning Phase

The Correlograph learning phase is made up of two steps [35].

1. The associative memory (matrix) $\boldsymbol{M}$ filled with values equal to zero is created.
2. It is subsequently updated according to the following expression:

$$m_{ij} = \begin{cases} 1 & if \quad y_i^{\mu} = 1 = x_j^{\mu}, \\ past\ value & \text{any other case.} \end{cases} \quad (5)$$

#### 4.2.2 Recovery Phase

The retrieval phase consists of presenting the previously trained associative memory $\boldsymbol{M}$ with an input vector $x^{\omega} \in A^n, A = \{0,1\}$. The way in which the input vector is presented to the associative memory is by making the product of the memory (matrix) $\boldsymbol{M}$ by the vector $x^{\omega}$. Subsequently, a thresholding operation is performed, according to the expression shown below [35]:

$$y_i^{\omega} = \begin{cases} 1 \quad if \quad \sum_{j=1}^{n} m_{ji} \cdot x_j^{\omega} \geq u, \\ 0 \qquad In\ any\ other\ case. \end{cases} \quad (6)$$

Likewise, $u$ is the threshold value, which its creators mention that an approximate estimate of its value is: $\log_2 n$, where *n* is equal to the dimension of the input patterns [35].

## 5 Proposed Algorithm

The proposed novel pattern classification algorithm, named Subtractive threshold associative classifier (STAC), belongs to the associative approach to pattern classification. Our proposed model is mainly based on the two pioneering associative memories, the Lernmatrix, which was created by Steinbuch, and the Correlograph, created by Willshaw.

In order for out proposed model to be able to deal with missing values and mixed data a preprocessing is applied to the dataset to resolve

this complexity in the data. On the other hand, the novel STAC algorithm makes use of an encoder, to convert real values to binary strings, as well as a mathematical transform.

The Johnson-Möbius method [36] and the $\tau^{[9]}$ transform [37] are explained. Johnson-Möbius encoder transfers all the values of the dataset with the purpose of eliminating negative values, in this sense, a sum of the minimum value is made within said set; Subsequently, if necessary, a number of decimals to be processed is set and, if required, the decimals are truncated so that they are adjusted to the set number of decimals; Afterwards, it is required to scale all the data in the set in order to eliminate these values.

Finally, to build the binary chain, the maximum number of the set is taken as a reference to define the length of the binary chain, where each real number is represented with as many ones as its value indicates, preceded by a string of zeros until the length is complete. defined.

On the other hand, the new STAC algorithm applies a process to transform the previously converted binary strings (using the Johnson-Möbius binary string encoder). This transformation uses a simple but powerful mathematical transform, called by the authors, the $\tau^{[9]}$ (Tau[9]) [37]. The $\tau^{[9]}$ transforms each binary component into a pair of binary values, based on the following expression:

$$\tau^{[9]}(1) = \binom{1}{0}, \quad \tau^{[9]}(0) = \binom{0}{1}. \tag{7}$$

The STAC algorithm consists of two phases, a learning phase and a retrieval (or classification) phase.

### 5.1 Learning Phase of STAC

1. All input patterns are converted to binary values using the Johnson-Möbius code.
2. A $\tau^{[9]}$ transform is applied to all the components of the input patterns converted in the step 1. The Tau[9] transform converts a binary digit into a pair of binary digits, according to the following: $\tau^{[9]}(1) = (1,0)$ and $\tau^{[9]}(0) = (0,1)$.
3. A one-hot output pattern is associated with each input pattern transformed in step 2.
4. The learning phase of the original Lernmatrix is performed in order to obtain the M matrix.

### 5.2 Classification Phase of STAC

1. The unknown input pattern $x^{\omega}$ is converted to binary values using the Johnson-Möbius code.
2. The $\tau^{[9]}$ transform (which was detailed in step 1 in the learning phase) is applied to all the components of the pattern converted in the step 1.
3. A value of $u$ is obtained, which is calculated as follows:

$$u = \log_2(\log_2 n + \sqrt{n})^3, \tag{8}$$

where, *n* is equal to the dimension of input patterns.

4. Using the $u$ value obtained in step 3, the recovery phase is performed from the original Lernmatrix, but modified according to the following expression:

$$y_i^{\omega} = \begin{cases} 1 & if \quad \sum_{j=1}^{n} m_{ij} \cdot x_j^{\omega} \geq umbral, \\ 0 & In\ any\ other\ case, \end{cases}$$
$$umbral = \left( \bigvee_{h=1}^{p} \left[ \sum_{j=1}^{n} m_{hj} \cdot x_j^{\omega} \right] \right) - u. \tag{9}$$

5. The proportions are voted according to the positions in each class corresponding to the $y^{\omega}$ pattern recovered in step 4, in order to obtain the predicted class of the unknown input pattern $x^{\omega}$.

## 6 Experimental Results and Discussion

This section reports experimental results using the proposal classifier STAC against the most relevant classifiers of state of art. On the other hands, section 6.1 describes the validation method and performance metrics.

Finally, section 6.2 presents classification results obtained by the algorithms.

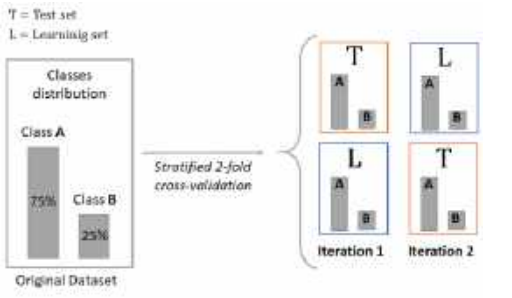



**Fig. 1.** Illustration of the stratified k-fold cross-validation method with k=2

### 6.1 Validation Method

In this section, we describe the validation method used in the experimental stage.

In order to obtain reliable results when measuring the performance of the classifiers in the experimentation stage, previously it is necessary to have implemented a validation method, which divides the original dataset into two sets: a test set and a learning set.

There are many ways to define these datasets, the most used and recommended by various authors is the k-fold cross-validation method [38]. However, because the datasets selected in the present research work mostly present class imbalance, it was decided to use the stratified 5x2 fold cross-validation (5x2 scv) method [26, 38], the which is widely recommended for imbalanced datasets, since they retain approximately the same percentage of patterns of each class for each of the folds.

In order to properly compare the classifiers executed in the experimental stage, it was necessary to apply a performance measure. Since the selected datasets present class imbalance, the Balanced Accuracy (BA) performance measure was used. This metric is recommended for imbalanced datasets by reason of it decreases the bias between the minority and majority class, thus obtaining results that reflect the true capacity of the classifiers [39].

The Balanced Accuracy metric for k classes is calculated as follows:

$$BA = \frac{1}{k}\sum_{i=1}^{k}\frac{T_i}{N_i}, \tag{10}$$

where, $T_i$ represents the number of correctly classified patterns of each class $i$, and $N_i$ represents the total number of patterns belonging to each class $i$.

### 6.2 Classification Results

Within the experimental phase that will be detailed below, two comparative experiments were carried out. The first shows the results obtained by the classifiers reported within the state of the art and the proposed STAC algorithm, which can be seen in Table 2, as well as the statistical results of these in Table 3 and Table 4.

The second experiment, shows the results obtained by the associative classifiers used as the basis for the STAC algorithm (Lernmatrix, Correlograph, and LM($\tau^{[9]}$), and the proposed model.

It can be seen from Table 2 that the proposed STAC algorithm achieved high performance, obtaining the best BA value on six of the twelve datasets used. As for example in the data sets: PostOperative, Lung-Cancer, ACPs Lung Cancer, Lymphography, Lymphography-NF and Primary-tumor.

In favor of our proposal, it can be noted that, in most cases, the performance values achieved by the proposed STAC classifier are close to the highest performances obtained by the other classifiers, such is the case of the set of Survey Lung-Cancer data, where the STAC algorithm obtained a result of *0.761*, which is not so far from the best performance, with a value of 0.779 achieved by the MLP classifier.

Other similar cases occur in the Newt-thyroid1, Newt-thyroid2 and Exasens dataset, where the proposed model resulted in performance values equal to 0.960, 0.969 and 0.902, respectively, very similar to the best performances obtained by the other classifiers.

To carry out a comparative analysis with greater reliability in the results, the Friedman test [7] was used, to demonstrate the existence of significant differences in the observed performances.

The proposed STAC model is placed at the first place in the ranking, with a value of *2.0417*, with respect to the remaining 5 algorithms, making it the

**Table 2.** Results obtained by state-of-the-art classifiers according to the BA measure

| Dataset | Naïve Bayes | 3-NN | MLP | SMV | C4.5 | STAC |
|---|---|---|---|---|---|---|
| PostOperative | 0.321 | 0.311 | 0.295 | 0.328 | 0.327 | **0.352** |
| Thyroid | 0.726 | 0.548 | 0.784 | 0.496 | **0.983** | 0.724 |
| Newt-thyroid1 | **0.988** | 0.913 | 0.965 | 0.745 | 0.926 | 0.960 |
| Newt-thyroid2 | **0.989** | 0.909 | 0.966 | 0.757 | 0.904 | 0.969 |
| Thoracic-Surgery | **0.578** | 0.508 | 0.523 | 0.500 | 0.511 | 0.508 |
| Lung-Cancer | 0.569 | 0.513 | 0.513 | 0.506 | 0.492 | **0.594** |
| Survey Lung-Cancer | 0.716 | 0.711 | **0.779** | 0.774 | 0.666 | 0.761 |
| ACPs Lung Cancer | 0.634 | 0.610 | 0.635 | 0.681 | 0.250 | **0.949** |
| Exasens COPD | 0.875 | 0.852 | 0.885 | 0.820 | **0.910** | 0.902 |
| Lymphography | 0.578 | 0.434 | 0.491 | 0.641 | 0.582 | **0.867** |
| Lymphography-NF | 0.747 | 0.498 | 0.793 | 0.698 | 0.598 | **0.965** |
| Primary-tumor | 0.271 | 0.230 | 0.234 | 0.252 | 0.233 | **0.340** |
| **Best BA** | 3 | 0 | 1 | 0 | 2 | **6** |

**Table 3**. Friedman test results

| Algorithm | Ranking[1] |
|---|---|
| STAC | 2.0417 |
| Naïve Bayes | 2.7500 |
| MLP | 3.0417 |
| C4.5 | 4.0000 |
| SMV | 4.1667 |
| 3-NN | 5.0000 |

[1]ordered from best to worst

**Table 4.** Post-hoc comparison obtained by the Holm test

| i | Algo | z | p | Holm Test |
|---|---|---|---|---|
| 5 | 3NN | 3.873 | 0.000 | 0.0033 |
| 4 | SMV | 2.782 | 0.005 | 0.0038 |
| 3 | C4.5 | 2.564 | 0.010 | 0.0045 |
| 2 | MLP | 1.309 | 0.190 | 0.0083 |
| 1 | NB | 0.927 | 0.353 | 0.0166 |

best model for the classification task described in this research work.

After performing the Friedman test, the null hypothesis is rejected with a confidence value of 95% and a probability value of $p = 0.001231$, which is largely below the level of significance established for this research, the which is $\alpha = 0.05$. Therefore, the existence of significant differences between the different classification algorithms is demonstrated.

Due to the results of the Friedman test, a post-hoc test was applied, the Holm test [8], the results of which can be seen in Table 3.

The test rejects the hypothesis with a value adjusted less than or equal to *0.05*.

**Table 5** Results obtained by the associative classifiers

| Dataset | CG | LM | LM($\tau^{[9]}$) | STAC |
|---|---|---|---|---|
| PostOperative | 0.200 | 0.318 | 0.255 | **0.352** |
| Thyroid | 0.000 | 0.297 | 0.625 | **0.724** |
| Newt-thyroid1 | 0.500 | 0.629 | 0.927 | **0.960** |
| Newt-thyroid2 | 0.500 | 0.624 | 0.961 | **0.969** |
| Thoracic-Surgery | 0.500 | **0.532** | 0.517 | 0.508 |
| Lung-Cancer | 0.333 | 0.386 | 0.553 | **0.594** |
| Survey Lung-Cancer | 0.500 | **0.816** | 0.674 | 0.761 |
| ACPs Lung Cancer | 0.250 | 0.856 | 0.941 | **0.949** |
| Exasens-COPD | 0.500 | 0.601 | 0.874 | **0.902** |
| Lymphography | 0.250 | 0.582 | 0.735 | **0.867** |
| Lymphography-NF | 0.500 | 0.79 | 0.814 | **0.965** |
| Primary-tumor | 0.047 | 0.055 | 0.282 | **0.340** |
| **Best BA** | 0 | 2 | 0 | **10** |

**Table 6.** Friedman test with the associative classifiers

| Algorithm | Ranking |
|---|---|
| STAC | 1.2500 |
| LM($\tau^{[9]}$) | 2.1667 |
| Lernmatrix | 2.5833 |
| Correlograph | 4.0000 |

**Table 7.** post-hoc by the Holm test on associative classifiers

| i | Algo | z | p | Holm |
|---|---|---|---|---|
| 3 | CG | 5.217 | 0.000 | 0.016 |
| 2 | LM | 2.529 | 0.011 | 0.025 |
| 1 | LM($\tau^{[9]}$) | 1.739 | 0.081 | 0.050 |

Therefore, it is observed that there are significant differences between the performances obtained by the proposed STAC algorithm and the classifiers: 3NN, SVM and C4.5.

After carrying out the experiments described in this section, it is observed that the proposed STAC model stood out with excellent results; because it significantly outperforms the other algorithms used in the state of the art under the same classification task.

In the second experiment, carried out with associative classification algorithms, of which the proposed STAC algorithm is based. Table 5 shows how the new proposed STAC algorithm clearly outperforms the other associative classifiers. Managing to obtain the best result in 10 of the 12 data sets used. To carry out a comparative analysis with greater reliability in the results, the Friedman test [39] was used, to demonstrate the existence of significant differences in the observed performances.

Table 6 shows the ranking obtained by the Friedman test according to the different associative classification algorithms presented. The proposed STAC model is placed at the first place in the ranking, with a value of 1.25, making it the best model for the classification task described in this present document.

After performing the Friedman test, the null hypothesis is rejected with a confidence value of 95% and a probability value of $p = 0.000003$, which is largely below the level of significance established for this research, the which is $\alpha = 0.05$. Therefore, the existence of significant differences between the different associative classifiers is demonstrated.

After performing the experiments described using the associative classifiers, it is observed that the proposed STAC algorithm stood out with competitive results; due to the significant differences between the performances obtained by the algorithm, obtained in two of the three associative classifiers used as a basis for the proposed STAC algorithm, under the same classification task.

Therefore, the results obtained support the statement that the proposal of the novel STAC model is suitable for the pre-diagnosis of the most common respiratory diseases.

## 7 Conclusion and Future Work

In the present work, a novel associative algorithm for pattern classification, STAC (Subtractive Threshold Associative Classifier) designed for the pre-diagnosis of respiratory diseases, was proposed and presented.

Likewise, another advantage of the STAC classifier is that it is an explainable model; making it transparent in its classification process, understanding why a pattern is classified to a certain class.

The experimental results carried out in section 6 point out the outstanding capacity of the proposed STAC algorithm, because they surpass several of the most used classification algorithms in the state of the art regarding the pre-diagnosis of respiratory diseases; excelling in exactly 6 of the 12 datasets used in the experimental phase.

Furthermore, according to the Friedman test, the best classifier in the experiments carried out was the STAC algorithm, indicating the presence of significant differences, with a probability value of p = 0.001230; Likewise, the post-hoc Holm test reflects that there is also the presence of significant differences in the performance obtained by the proposed algorithm and the other classifiers.

In future work, the intention will be to apply the novel STAC algorithm on datasets with different approaches, with the aim of evaluating its performance and behavior in different diseases or even non-medical datasets; likewise, it is proposed to compare the STAC algorithm with more state-of-the-art classifiers.

Finally, it is planned to prove why the proposed threshold works properly, and considerably improves the performance of the algorithm STAC. With this, it is proposed to consider some more in-depth analysis on the behavior of the model when using different threshold and the one proposed in this work.

## Acknowledgments

The authors would like to thank the Instituto Politécnico Nacional (Secretaría Académica, SIP, CIDETEC, and CIC), the CONACyT, and SNI for their economic support to develop this work.

## References

1. **Abdar, M., Zomorodi-Moghadam, M., Das, R., Ting, I. H. (2017).** Performance analysis of classification algorithms on early detection of liver disease. Expert Systems with Applications, Vol. 67, pp. 239–251. DOI: 10.1016/j.eswa.2016.08.065.
2. **Woloshin, S., Patel, N., Kesselheim, A. S. (2020).** False negative tests for SARS-CoV-2 infection-challenges and implications. New England Journal of Medicine, Vol. 383, No. 6, p. e38. DOI: 10.1056/NEJMp2015897.
3. **National Cancer Institute (2022).** Respiratory disease. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/respiratory-disease
4. **Forum of International Respiratory Societies (2021).** The global impact of respiratory disease, Third Edition, European Respiratory Society, firsnet.org/images/publications/FIRS_Master_09202021.pdf.
5. **Sengupta, N., Sahidullah, M., Saha, G. (2016).** Lung sound classification using cepstral-based statistical features. Computers

in Biology and Medicine, Vol. 75, pp. 118–129. DOI: 10.1016/j.compbiomed.2016.05.013.

6. **World Health Organization (2022).** WHO coronavirus (COVID-19) dashboard. https://covid 19. who.int
7. **World Health Organization (2022).** The top 10 causes of death. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death
8. **Instituto Nacional de Estadística y Geografía (INEGI) (2022).** Estadística de defunciones registradas de Enero a Junio de 2021 (Preliminar). https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2022/dr/dr2021.pdf.
9. **Doi, K. (2007).** Computer-aided diagnosis in medical imaging: historical review, current status and future potential. Computerized Medical Imaging and Graphics, Vol. 31, No. 4-5, pp. 198–211. DOI: 10.1016/j.compmedimag.2007.02.002.
10. **Luján-García, J. E., Yáñez-Márquez, C., Villuendas-Rey, Y., Camacho-Nieto, O. (2020).** A transfer learning method for pneumonia classification and visualization. Applied Sciences, Vol. 10, No. 8, p. 2908. DOI: 10.3390/app10082908.
11. **Narin, A., Kaya, C., Pamuk, Z. (2021).** Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. Pattern Analysis and Applications, Vol. 24, pp. 1207–1220. DOI: 10.1007/s10044-021-00984-y.
12. **Wolpert, D. H., Macready, W. G. (1997).** No free lunch theorems for optimization. IEEE transactions on evolutionary computation, Vol. 1, No. 1, pp. 67–82. DOI: 10.1109/4235.585893.
13. **Estapé, J. V., Zboromyrska, Y., Gómez, A. V., Cancho, I. A., García, E. R., Álvarez-Martínez, M. J., Maeso, M. Á. M. (2016).** Métodos moleculares de diagnóstico de infecciones respiratorias. ¿Ha cambiado el esquema diagnóstico? Enfermedades Infecciosas y Microbiología Clínica, Vol. 34, pp. 40–46. DOI: 10.1016/S0213-005X(16)30218-X.
14. **Maleki, N., Zeinali, Y., Niaki, S. T. A. (2021).** A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection. Expert Systems with Applications, Vol. 164, pp. 113981. DOI: 10.1016/j.eswa.2020.113981.
15. **Acevedo-Mosqueda, M. E., Yáñez-Márquez. C. (2006).** Alpha-beta bidirectional associative memories [Memorias asociativas bidireccionales alfa-beta]. Computación y Sistemas, Vol. 10, No. 1, pp. 82–90.
16. **Cardoso, I., Almeida, E., Allende-Cid, H., Frery, A. C., Rangayyan, R. M., Azevedo-Marques, P. M., Ramos, H. S. (2018).** Analysis of machine learning algorithms for diagnosis of diffuse lung diseases. Methods of Information in Medicine, Vol. 57, No. 05–06, pp. 272–279. DOI: 10.1055/s-0039-1681086.
17. **Finkelstein, J., Jeong, I. C. (2017).** Machine learning approaches to personalize early prediction of asthma exacerbations. Annals of the New York Academy of Sciences, Vol. 1387, No. 1, pp. 153–165. DOI: 10.1111/nyas.13218.
18. **Amaral, J. L., Lopes, A. J., Veiga, J., Faria, A. C., Melo, P. L. (2017).** High-accuracy detection of airway obstruction in asthma using machine learning algorithms and forced oscillation measurements. Computer Methods and Programs in Biomedicine, Vol. 144, pp. 113–125. DOI: 10.1016/j.cmpb.2017.03.023.
19. **Radhika, P. R., Nair, R. A., Veena, G. (2019,).** A comparative study of lung cancer detection using machine learning algorithms. 2019 IEEE International Conference on Electrical, Computer and Communication Technologies, IEEE, pp. 1–4. DOI: 10.1109/ICECCT.2019.8869001.
20. **Gonem, S., Janssens, W., Das, N., Topalovic, M. (2020).** Applications of artificial intelligence and machine learning in respiratory medicine. Thorax, Vol. 75, No. 8, pp. 695–701. DOI: 10.1136/thoraxjnl-2020-214556
21. **Xiong, Y., Ba, X., Hou, A., Zhang, K., Chen, L., Li, T. (2018).** Automatic detection of mycobacterium tuberculosis using artificial intelligence. Journal of Thoracic Disease, Vol.

10, No. 3, p. 1936. DOI: 10.21037/jtd.2018.01.91.

22. **Christe, A., Peters, A. A., Drakopoulos, D., Heverhagen, J. T., Geiser, T., Stathopoulou, T., Ebner, L. (2019).** Computer-aided diagnosis of pulmonary fibrosis using deep learning and CT images. Investigative Radiology, Vol. 54, No. 10, p. 627. DOI: 10.1097/RLI.0000000000000574.
23. **Soldati, G., Demi, M., Smargiassi, A., Inchingolo, R., Demi, L. (2019).** The role of ultrasound lung artifacts in the diagnosis of respiratory diseases. Expert Review of Respiratory Medicine, Vol. 13, No. 2, pp. 163–172. DOI: 10.1080/17476348.2019.1565997.
24. **Pham, L., McLoughlin, I., Phan, H., Tran, M., Nguyen, T., Palaniappan, R. (2020).** Robust deep learning framework for predicting respiratory anomalies and diseases. 2020 42nd annual international conference of the IEEE engineering in medicine & biology society, pp. 164–167, DOI: 10.1109/EMBC44109.2020.9175704.
25. **Alcalá-Fdez, J., Fernádez, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera, F. (2011).** Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. Journal of Multiple-Valued Logic and Soft Computing, Vol. 17, No. 2–3, pp. 255–287.
26. **Dua, D., Graff, C.** (University of California, Irvine, School of Information). UCI Machine Learning Repository.
27. **Sossa-Azuela, J. H., Yañez-Marquez, C. (2001).** Computing geometric moments using morphological erosions. Pattern Recognition, Vol. 34, No. 2, pp. 271–276. DOI: 10.1016/S0031-3203(99)00213-7.
28. **López-Yáñez, I., Sheremetov, L., Yáñez-Márquez, C. (2014).** A novel associative model for time series data mining. Pattern Recognition Letters, Vol. 41, pp. 23–33. DOI: 10.1016/j.patrec.2013.11.008.
29. **De-la-Vega, A. R. D., Villuendas-Rey, Y., Yáñez-Márquez, C., Camacho-Nieto, O. (2020).** The Naïve associative classifier with epsilon disambiguation. IEEE Access, Vol. 8, pp. 51862–51870. DOI: 10.1109/ACCESS.2020.2979054.
30. **Lytras, M. D., Mathkour, H., Abdalla, H. I., Yáñez-Márquez, C., de-Pablos, P. O. (2014).** The social media in academia and education: research R-evolutions and a paradox: advanced next generation social learning innovation. Journal of Universal Computer Science, Vol. 20, No. 15, pp. 1987–1994.
31. **Widrow, B., Lehr, M. A. (1990).** 30 years of adaptive neural networks: perceptron, madaline, and backpropagation. Proceedings of the IEEE, Vol. 78, No. 9, pp. 1415–1442. DOI: 10.1109/5.58323.
32. **Platt, J. C. (1998).** Sequential minimal optimization: A fast algorithm for training support vector machines. MSRTR: Microsoft Research, Vol. 3, No. 1, pp. 88–95.
33. **Quinlan, J. R. (2014).** C4.5: programs for machine learning. Elsevier.
34. **Steinbuch, K. (1961).** Die lernmatrix. Kybernetik, Vol. 1, pp. 36–45. DOI: 10.1007/BF00293853.
35. **Yáñez-Márquez, C., López-Yáñez, I., Aldape-Pérez, M., Camacho-Nieto, O., Argüelles-Cruz, A. J., Villuendas-Rey, Y. (2018).** Theoretical foundations for the alpha-beta associative memories: 10 years of derived extensions, models, and applications. Neural Processing Letters, Vol. 48, pp. 811–847. DOI: 10.1007/s11063-017-9768-2.
36. **Papadomanolakis, K. S., Kakarountas, A. P., Sklavos, N., Goutis, C. E. (2002).** A fast Johnson-Mobius encoding scheme for fault secure binary counters. Proceedings of Design, Automation and Test in Europe, p. 1.
37. **Velazquez-Rodriguez, J. L., Villuendas-Rey, Y., Camacho-Nieto, O., Yanez-Marquez, C. (2020).** A novel and simple mathematical transform improves the perfomance of lernmatrix in pattern classification. Mathematics, Vol. 8. No. 5, p. 732. DOI: 10.3390/math8050732.
38. **Nakatsu, R. T. (2020).** An evaluation of four resampling methods used in machine learning classification. IEEE Intelligent Systems, Vol. 36, No. 3, pp. 51–57.

**39. Dietterich, T. G. (1998).** Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation, Vol. 10, No. 7, pp. 1895–1923. DOI: 10.1162/089976698300017197.

*Article received on 24/04/2024; accepted on 06/06/2024.*
**Corresponding author is Yenny Villuendas-Rey.*

ISSN 2007-9737

# Cyberbullying Detection in a Multi-classification Codemixed Dataset

Sahinur Rahman-Laskar[1,*], Gauri Gupta[1], Ritika Badhani[1], David Eduardo Pinto-Avendaño[2]

[1] UPES, School of Computer Science,
India

[2] Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación,
Mexico

{sahinurlaskar.nits, gauri17gupta, ritika.badhani12, davideduardopinto}@gmail.com

**Abstract.** In an era characterized by digital communication and social media, the concept of cyberbullying has arisen as a social concern, impacting individuals of all ages. It refers to the act of using digital communication tools like, social media, and messaging apps, to harass intimidate or harm someone. Codemixed cyberbullying refers to the use of multiple languages or a mix of languages in online communications and the use of multiple languages or a mix of languages can sometimes make it challenging for content moderators or automated systems to detect and address cyberbullying effectively. The challenges include the availability of standard codemixed datasets, especially for Indian languages. This paper investigates cyberbullying detection in Hinglish, a code-mixed language of Hindi and English. We have created a novel multi-class Hinglish dataset, annotated across seven cyberbullying categories: age, gender, religion, mockery, abusive, offensive, and not cyberbullying, and explored different machine learning-based models. We have performed a comparative analysis based on the standard evaluation metrics and achieved a state-of-the-art result on a multi-class codemixed Hinglish dataset.

**Keywords.** Cyberbullying, codemixed, Hinglish, machine learning.

## 1 Introduction

Cyberbullying has emerged as a pressing concern in the digital era, particularly on online communication platforms.

It manifests in various forms, such as threats, hate speech, and harassment, inflicting detrimental effects on victims [28]. Identifying and addressing cyberbullying is imperative for creating a secure and inclusive online environment for all users. Detecting cyberbullying involves locating instances of the phenomenon in online content, such as messages, comments, and social media posts. Natural language processing, sentiment analysis, and artificial intelligence techniques offer promising approaches for this task [20].

The primary objectives are to accurately identify cyberbullying incidents and classify them as threats, hate speech, and harassment. Machine learning models have demonstrated remarkable potential in cyberbullying detection by discerning patterns from large datasets, a task that can be challenging for humans. The initial crucial step in developing a cyberbullying detection system is acquiring a comprehensive and diverse dataset of online content, including labeled instances of various cyberbullying forms.

This dataset should encompass a representative sample of threats, hate speech, harassment, and other cyberbullying manifestations. Data preprocessing follows, involving cleaning, deduplicating, and formatting the data into a machine-learning-compatible format.

Relevant features, such as part-of-speech tagging, sentiment scores, and word frequencies, are then extracted. Subsequently, machine learning algorithms are employed to build and train models on the preprocessed data and extracted

**Table 1.** Data statistics of train and test set

| Split | No. of Comments |
|---|---|
| Train | 7400 |
| Test | 1000 |
| Total | 8400 |

features. The models' performance is evaluated using a separate dataset of online content, considering metrics such as recall, accuracy, and F1-score.

Detecting cyberbullying in code-mixed languages like Hinglish poses additional challenges due to the potential ambiguity in meaning [25]. However, developing models capable of accurately identifying cyberbullying in multilingual and code-mixed contexts is crucial for promoting a safer and more inclusive online environment.

Identifying cyberbullying is a critical step towards providing support and resources to victims and perpetrators alike. Such resources may include counseling services, online forums, and educational materials aimed at mitigating the detrimental effects of cyberbullying and fostering a more positive online experience.

In summary, detecting cyberbullying through machine learning techniques is an essential endeavor in creating a secure and welcoming cyberspace. Developing accurate and efficient models can enable the timely identification of cyberbullying incidents [5]. Providing support and resources to victims is paramount to alleviating the harmful impacts of cyberbullying and promoting a more conducive online environment. The key contributions to this work are as follows:

- Created a multi-classification Hinglish code-mixed dataset, namely, MC-Hinglish$1.0$ which will be publicly available[1].
- Performed a comparative analysis by exploring different machine learning models for cyberbullying detection on the developed Hinglish code-mixed dataset.
- Proposed the use of an ensemble model and achieved state-of-the-art results for cyberbullying detection on the multi-classification Hinglish code-mixed dataset.

The rest of the paper is organized as follows. In Section 2, existing works related to cyberbullying and the relevant work on Hinglish cyberbullying detection are discussed briefly. The preparation of the dataset MC-Hinglish$1.0$ is described in Section 3. The ensemble model to combine the different machine learning models, namely, Support Vector Machines (SVM), Random Forest, Logistic Regression, Multinomial Naive Bayes, and XGBoost, are presented in Section 4. Section 5 presents experimental results and Analysis and Section 6, concludes the paper with future work.

## 2 Related Work

Fuzzy logic and multinomial Naive Bayes classification are suggested in [3] as ways to identify various forms of cyberbullying in Facebook comments. Before acquiring characteristics like adjective and noun frequency, preprocessing procedures like tokenization and stopword elimination are used. The classifier recognizes various forms of bullying, including shame, racism, and sexual harassment.

Bullying severity is assessed by fuzzy rules based on age and the number of expletives. The model outperforms an SVM technique, achieving 88.76% accuracy on a benchmark dataset. In order to detect cyberbullying across languages, the paper shows how to modify natural language processing (NLP) approaches such as Naive Bayes. Bullying severity nuances are captured by fuzzy logic. Both culture detection and model accuracy can be enhanced by additional work.

The research in [9] looks into ways to make it easier to spot instances of cyberbullying on social media sites like Twitter. The research expands on what is already known about the characteristics, trends, and detection strategies of cyberbullying.

The first section of the study discusses the pervasiveness of cyberbullying and emphasizes its negative impacts, especially on vulnerable populations like adolescents. It highlights the

[1] github.com/sahinurlaskar/MC-Hinglish1.0

**Table 2.** Class level data statistics

| Classes | Age | Gender | Religion | Offensive | Mockery | Abusive | Not-Cyberbullying |
|---|---|---|---|---|---|---|---|
| **Train** | 977 | 1004 | 979 | 979 | 980 | 989 | 1491 |
| | 143 | 141 | 127 | 128 | 135 | 115 | 211 |
| | 1120 | 1145 | 1106 | 1107 | 1115 | 1104 | 1702 |
| **Test** | 129 | 141 | 131 | 124 | 127 | 148 | 200 |

significance of social media sites like Twitter, which help incidences of cyberbullying spread quickly. The researchers use a variety of data, such as thoughts, emotions, and Twitter-specific traits, to enhance detection.

User personalities have been established on the Big Five and Dark Triad models. The process of extracting pertinent features, including text/content, users, and information about networks, from Twitter data is described in the paper. It explores the relationship between Dark Triad constructions and the Big Five personality traits and talks about how these characteristics might be used to spot cyberbullying activities.

To comprehend the subtleties of the discourse around cyberbullying, the research also investigates the application of emotion and sentiment analysis. It draws attention to the shortcomings of the methods used now and makes suggestions for how to get around them. In summary, by combining innovative features and utilizing cutting-edge analytical methods, our work advances the creation of reliable and precise cyberbullying detection systems.

It hopes to accomplish this by offering insightful information on the intricate dynamics of bullying and assisting in the development of safer online spaces. The crucial issue of identifying cyberbullying in the Indian languages of Marathi and Hindi is the subject of study in [24]. Cyberbullying, or the act of repeatedly injuring victims via the use of digital media, has grown to be a significant social problem. The authors give background information on the emergence of cyberbullying, the harm it causes to victims' physical and emotional health, particularly young people, and the necessity of automatic detection systems. They claim that despite the large number of non-English-speaking users on social media, the majority of detection research focuses on English material.

According to the authors' survey of related work, the majority of machine learning research on textual features used for cyberbullying identification is conducted for English language users. Naive Bayes, SVM, decision tree models, and logistic regression are among the approaches used to conduct promising research in other languages, such as Arabic and Turkish. Sentiment analysis, network patterns, and keywords are important characteristics. BullyBlocker is a technology that recognizes and notifies parents of cyberbullying on Facebook.

The authors [24] gather datasets in Hindi and Marathi from a variety of sources, including newspapers, reviews, and social media. Bullying is manually classified in the data. Synthetic oversampling is employed to balance classrooms because bullying accounts for just 9% of texts. Bag-of-words feature extraction is carried out following preprocessing.

A split of the data into 80/20 is used for training and testing various models, such as logistic regression, stochastic gradient descent, and multinomial Naive Bayes. Based on synthetic data, experiments indicate that logistic regression performs best, achieving up to 97% accuracy and 96% F1 score. This method works for all languages and domains. Larger datasets will be used in future research, along with linguistically specific sentiment analysis, comparisons with NLP techniques, and real-time social media integration.

The study in paper [22] suggested utilizing classifiers like Naive Bayes and Random Forest with psychological variables like sentiment and personality of Twitter users to increase

**Table 3.** Class level data statistics

| Class Name | Example |
|---|---|
| AGE | Behan meri 40 yr maid looks better than you |
| GENDER | Electric scotty chalane vale ladke gay hote h |
| RELIGION | Na pakistani pta paye na Indian, na angrez |
| MOCKERY | Arey bc ye icon kya hai yaar, tujhe itna bhi nahi pata |
| OFFENSIVE | Dettol ke add m kitanu ka role krogi |
| ABUSIVE | Bc mene teko call bhi kiya tha |
| NOT_CYBERBULLYING | Nai na, bahut garmi ho raha hai |

cyberbullying detection accuracy to 91.88%. However, this strategy is limited by the need for sentiment data.

By combining contextual embeddings from approaches such as BERT and VecMap, [18] created a multitask framework that achieved over 80% accuracy in sentiment analysis and cyberbullying detection. More study is necessary because cyberbullying is multimodal. Numerous research works have explicitly addressed the Bangla/Bengali language.

Using CNN models, [2] obtained 84% and 80% accuracy for the Bangla and Romanized Bangla datasets. With just 1,339 comments, [1] created SVM and Naive Bayes models that achieved up to 72% accuracy. [13] discovered a maximum accuracy of 78.1% when investigating several ML algorithms using TF-IDF characteristics.

[27] compared LSTM and CNN to ML models like SVM and Naive Bayes using data from Bangla YouTube comments. The accuracy of the LSTM using word embeddings was 65–67%. In order to identify abusive Bangla comments in Facebook postings based on text and emojis, [11] examined MNB, SVM, and CNN-LSTM models; SVM proved to be the most successful with 78% accuracy. Larger datasets and more intricate deep learning architectures require further investigation.

With the help of a dataset of 12,282 social media comments, this study creates a Bi-LSTM model with two thick layers for the identification of cyberbullying in Bangla. Model training precedes the application of preprocessing techniques like tokenization, stemming, and padding sequences.

The Adam optimizer yields a higher testing accuracy of 95.08% when compared to the SGD optimizer's performance. 94.31% average accuracy is achieved with five-fold cross-validation. A benefit over earlier work is a bigger dataset and less feature engineering. Techniques like attention processes can still be used to improve accuracy.

The research in [8] provides a sentiment analysis approach to identify cyberbullying in tweets by utilizing machine learning techniques. The authors gather a corpus of tweets and manually label them as neutral, negative, or positive comments.

The tweets undergo several preparation procedures, such as cleaning, tokenization, removal of stop words, normalization, named-entity recognition, and stemming. The preprocessed tweets are used to extract N-gram characteristics ranging from $2$ to $4$ grams. Chi-square feature selection is used, and information acquired is utilized to eliminate features that aren't important.

Accuracy, precision, recall, F1-score, and ROC curve are used to evaluate the Naive Bayes (NB) and Support Vector Machine (SVM) classifiers after they have been trained on the features. Tests reveal that SVM beats NB for every n-gram model, with 4-grams exhibiting the greatest results.

The suggested model results in a somewhat better detection of cyberbullying than earlier research, with SVM achieving 91.64% accuracy and 88.93% ROC as opposed to 83.46% and

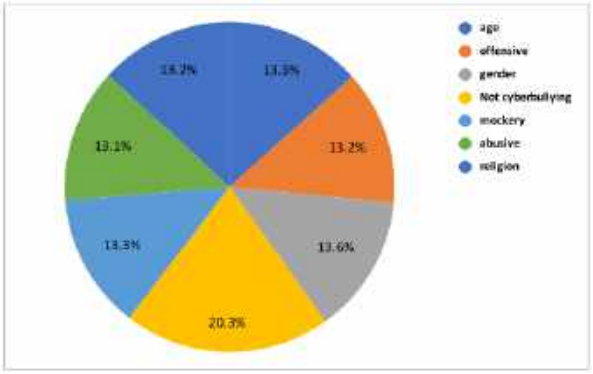



**Fig. 1.** Train data distribution

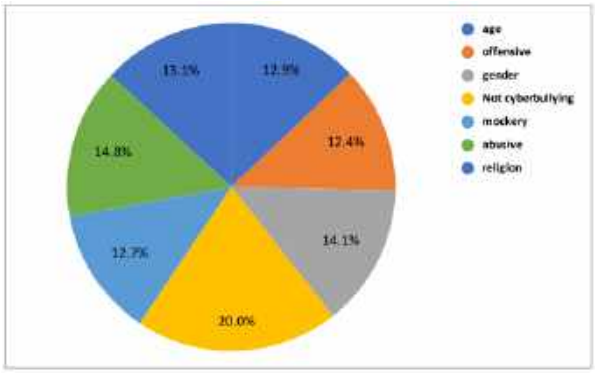



**Fig. 2.** Test data distribution

85.71% from earlier research. Overall, the sentiment analysis model outperforms both NB and earlier methods in the efficient real-time identification of cyberbullying tweets. It uses an SVM classifier with 4-gram features.

Benchmark research comparing multiple neural networks and machine learning models for the binary categorization of cyberbullying in social media texts is presented in this publication [29]. The authors make use of seven real-world datasets that are currently available from various social media sites, such as Twitter, Instagram, Vine, Ask.fm, Formspring, and Ask.fm. Following preprocessing and data cleaning, over $390,000$ sentences were classified as either non-cyberbullying or cyberbullying in the combined datasets. The authors utilize random oversampling to correct for class imbalance in the datasets. Four models are compared: a Support Vector Machine (SVM) baseline, a BiLSTM network, and tuned versions of the BERT and HateBERT transformer models. For two to four training epochs, hyperparameters are adjusted. After being trained on combined datasets from several platforms, the models are assessed on holdout test sets from each platform.

The development of cyberbullying detection models that are indifferent to platforms is initiated by this cross-platform assessment. Based on test sets that are matched, the optimized HateBERT model outperforms the others overall, with F1 scores as high as 0.81 and as low as 0.69–0.76 on sets that are mismatched. The maximal F1 scores of the SVM and BiLSTM models are approximately 0.70. Applying models across platforms significantly reduces performance. The authors draw the conclusion that HateBERT has potential for widespread cyberbullying detection.

Seven machine learning classifiers for cyberbullying detection on Twitter are compared in the work [21]. A dataset including 37,373 tweets—70% for training and 30% for prediction—is used in the study. The usage of TF-IDF and Word2Vec algorithms, as well as the significance of pre-processing and feature extraction in cyberbullying detection, are covered in the paper. The study contrasts the effectiveness of several classifiers, including Support Vector Machine, Random Forest, and Naive Bayes. using an accuracy of 95.5%, the SVM classifier using a linear kernel outperforms the others, according to the results. The study comes to the conclusion that machine learning methods can be useful in identifying harassment on social media sites, and the suggested model can serve as a foundation for further studies in this field.

### 2.1 Existing Work on Hinglish Codemixed Dataset

In order to detect offensive content in tweets in paper [17] investigate the use of a variety of classifiers, including Naive Bayes and Support Vector Machine. It also explores the stages of feature extraction and preprocessing, emphasizing how crucial it is to extract pertinent features and clean text in order to achieve precise classification.

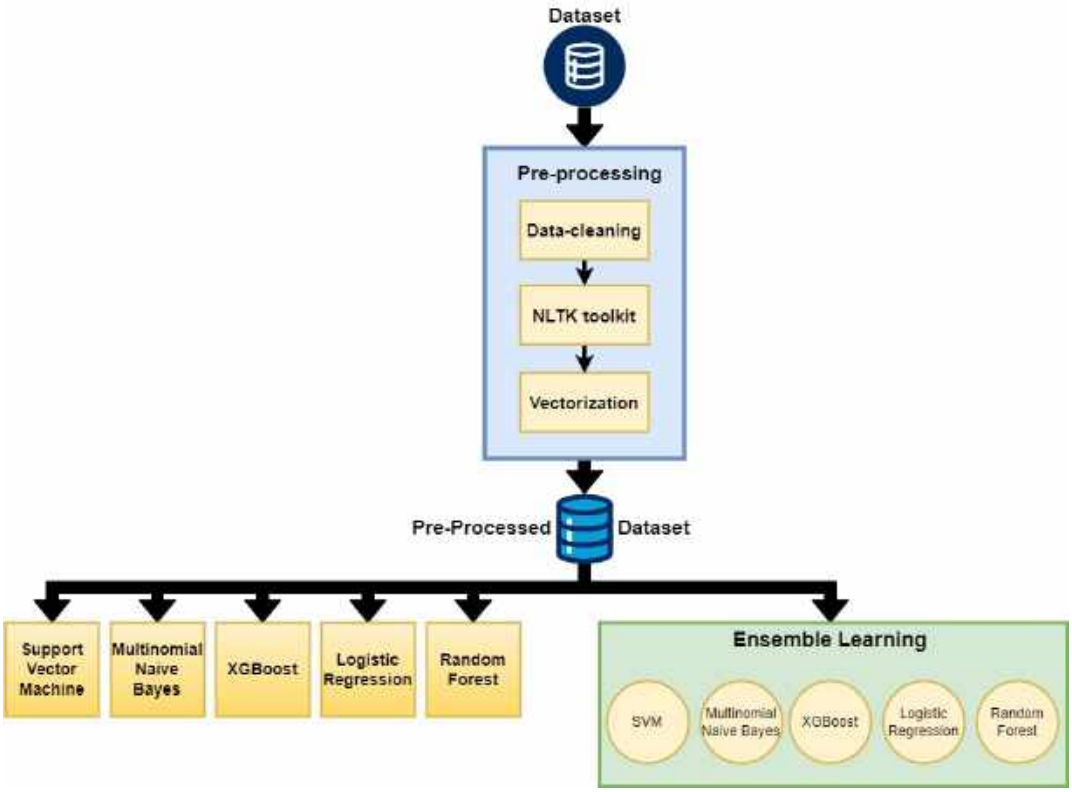



**Fig. 3.** Methodology

The work uses a data set of 37,373 tweets to assess the effectiveness of several classifiers and methods for extracting features, and in the end, it suggests a model for social media platform cyberbullying detection.

BullyExplain is a unique benchmark dataset for explainable cyberbullying detection in code-mixed language, and it is presented in the publication [16]. The multitask issue is reframed as a text-to-text creation task using the authors' novel unified generative framework, GenEx. Applied to the BullyExplain dataset, the suggested method outperforms other baseline models and existing state-of-the-art methods in a number of evaluation measures.

The study draws attention to the frequency of code-mixing in cyberbullying as well as the necessity of offering explanations for machine learning decisions. There are 6,084 samples in the dataset overall; 3,034 of them are classified as non-bully, and the remaining 3,050 samples are classified as bullies. A commonsense-aware unified generative framework called GenEx is also introduced by the authors. It uses commonsense information to improve the context and richness of tweets that are usually concise and to the point.

The study in [12] addresses the challenge of detecting cyberbullying in Hinglish, a code-mixed language combining Hindi and English. They created a new dataset called "CMDL-Cyberbullying" with 20,000 manually annotated Hinglish comments from social media platforms. Using this dataset, they experimented with various machine learning models like logistic regression, naive Bayes, decision trees, random forest, and support vector machines (SVMs) for cyberbullying classification. Different features such as n-grams, TF-IDF, and pre-trained word embeddings were evaluated. The SVM model with character n-grams and TF-IDF features performed

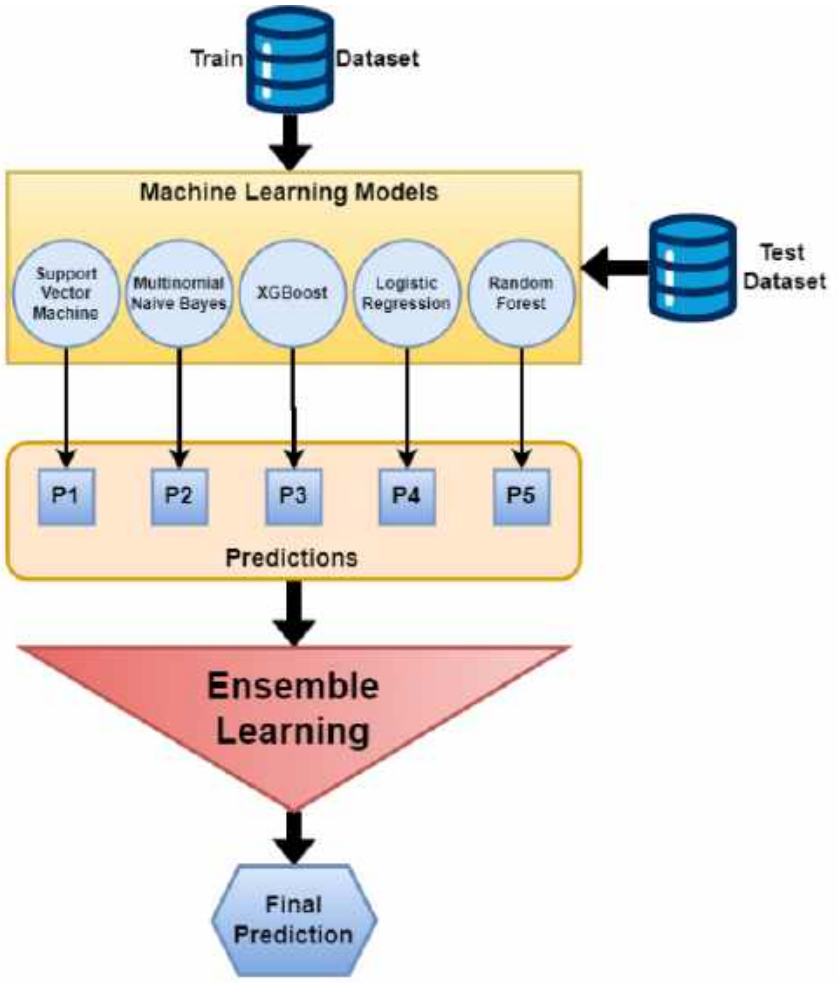



**Fig. 4.** Ensemble learning model

best, achieving an F1-score of 0.78. However, the presence of English words, phonetic typing of Hindi, and Romanized Hindi posed challenges.

The authors highlight the importance of detecting cyberbullying in code-mixed languages and the need for further research to improve model performance in such scenarios. The paper [19] proposes a novel approach to detect cyberbullying in Hinglish (Hindi-English code-mixed) text by leveraging emojis, sentiment analysis, and emotion detection. The authors created a new dataset called ”HingC” containing 16,988 Hinglish comments annotated for cyberbullying, sentiment, emotions, and emojis.

They developed an ensemble model that combines features derived from emojis, sentiment scores, emotion scores, and text representations. For text representation, they used multilingual BERT and contextual string embeddings from the Flair library. Emoji representations were obtained from the DeepMoji model. The sentiment and emotion analysis components utilized transfer learning from English to Hinglish.

The ensemble model, which combined all these features, achieved state-of-the-art performance on the HingC dataset with an F1 score of 0.81 for cyberbullying detection. The authors demonstrated that incorporating emojis, sentiment, and emotion

**Table 4.** Experimental result

| | Logistic Regression | Random Forest | XGBoost | Naive Bayes | SVM | Voting(Hybrid) |
|---|---|---|---|---|---|---|
| **Accuracy** | 0.580 | 0.570 | 0.602 | 0.512 | 0.564 | 0.609 |
| **Recall** | 0.5643 | 0.5619 | 0.5908 | 0.4963 | 0.5524 | 0.5940 |
| **F1-score** | 0.5540 | 0.5585 | 0.5804 | 0.4731 | 0.5475 | 0.5879 |

information significantly improved the cyberbullying detection performance compared to using only text features. The paper highlights the importance of these multimodal signals for understanding the nuances of code-mixed languages like Hinglish in cyberbullying contexts.

This paper [10] focuses on developing an efficient model to detect the presence of Hinglish text (a code-mixed language of Hindi and English) in YouTube data, such as video titles, descriptions, and comments. The authors highlight the importance of identifying Hinglish text for tasks like content moderation, targeted advertising, and understanding user preferences. The proposed model uses a combination of rule-based and machine learning approaches.

First, a set of rules are applied to filter out text that is purely in English or Hindi. For the remaining text, which potentially contains Hinglish, character-level and word-level features are extracted. These features include n-grams, distribution of Hindi and English characters, presence of named entities, and part-of-speech tags.

The extracted features are then fed into several machine learning classifiers, including Logistic Regression, Support Vector Machines (SVMs), and ensemble methods like Random Forest and XGBoost. The models are trained and evaluated on a manually annotated dataset of YouTube data. The authors report that their best performing model, an ensemble of XGBoost and Logistic Regression, achieved an F1-score of 0.93 in detecting the presence of Hinglish text.

They also conducted experiments to analyze the impact of different features and found that character-level n-grams and the distribution of Hindi and English characters were the most important features for this task. The paper [17] proposes a novel generative approach for explainable cyberbullying detection in Hinglish (Hindi-English code-mixed) text.

The authors use a variational autoencoder (VAE) model to learn disentangled latent representations that capture different attributes like cyberbullying, sentiment, and emotions present in the input text. During inference, they generate counterfactual samples by manipulating these latent representations to analyze how changes in attributes like sentiment impact the cyberbullying probability. This provides explainable insights into the model's predictions.

On their curated HingC dataset, their VAE-based model achieved state-of-the-art F1 score of 0.82 for cyberbullying detection, while also enabling explainable AI capabilities lacking in previous black-box methods. The authors demonstrate how their approach can highlight words/phrases responsible for cyberbullying flags and generate revised non-cyberbullying versions through latent edits, aiding transparency and safer content generation.

However, it is evident from the reviewed literature that there is a significant lack of research focusing on the development and utilization of multi-classified Hinglish datasets, which is a unique blend of Hindi and English languages widely used in various online platforms and social media. This research gap underscores the need for further exploration and construction of robust multi-classified Hinglish datasets to facilitate more effective and accurate cyberbullying detection models tailored to the diverse linguistic landscape of the Indian subcontinent.

## 3 Dataset Creation: MC-Hinglish 1.0:

The dataset utilized in this paper was created to facilitate the task of cyberbullying detection in the Hinglish language. Initially, a binary classified

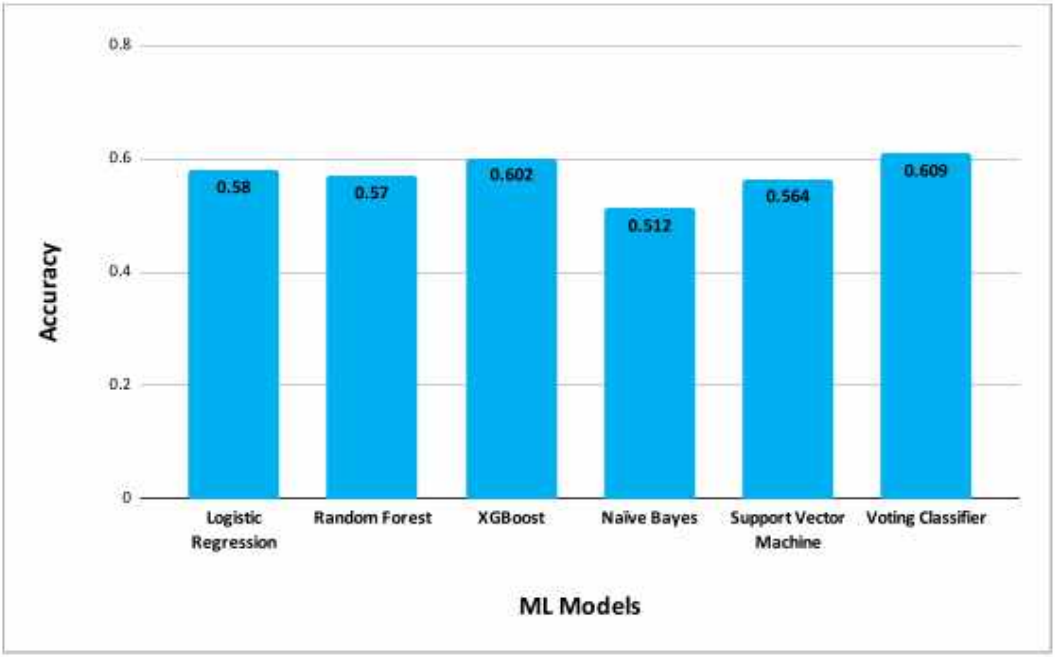



**Fig. 5.** Accuracy graph

dataset was obtained from GitHub[2], consisting of comments labeled as either cyberbullying or non-cyberbullying.

Subsequently, this dataset was manually annotated and modified to transform into a multi-classification dataset with seven distinct categories: age, gender, religion, mockery, abusive, offensive, and not-cyberbullying. The annotation process aimed to provide a more granular understanding of the types of cyberbullying present in the dataset.

### 3.1 Data Collection

The original binary dataset was sourced from the aforementioned GitHub repository. This dataset contained a collection of comments written in Hinglish language, spanning Twitter.

Prior to annotation, basic preprocessing steps are applied to clean the text data and remove unwanted symbols.

[2]http://surl.li/trwil

### 3.2 Data Statistics

MC-Hinglish$1.0$ dataset comprises a total of $8400$ annotated comments, with $7400$ comments allocated for the training set, and 1000 comments reserved for the testing set. The dataset (MC-Hinglish 1.0) was split into training, and test sets in the ratio of $88:12$.

Care was taken to maintain the class distribution across the splits to prevent any bias in model training and testing (as shown in Fig. 1, 2). The data statistics are presented in Table 1.

### 3.3 Class Description

The manual annotation process involved categorizing each comment into one of the seven predefined classes based on its content. The manually annotated dataset (MC-Hinglish$1.0$) is categorized into $7$ classes: age, gender, religion, mockery, abusive, offensive, and not-cyberbullying. This annotation process took approximately $60$ days of rigorous effort and ensured no duplicate comments. Each comment is labeled according

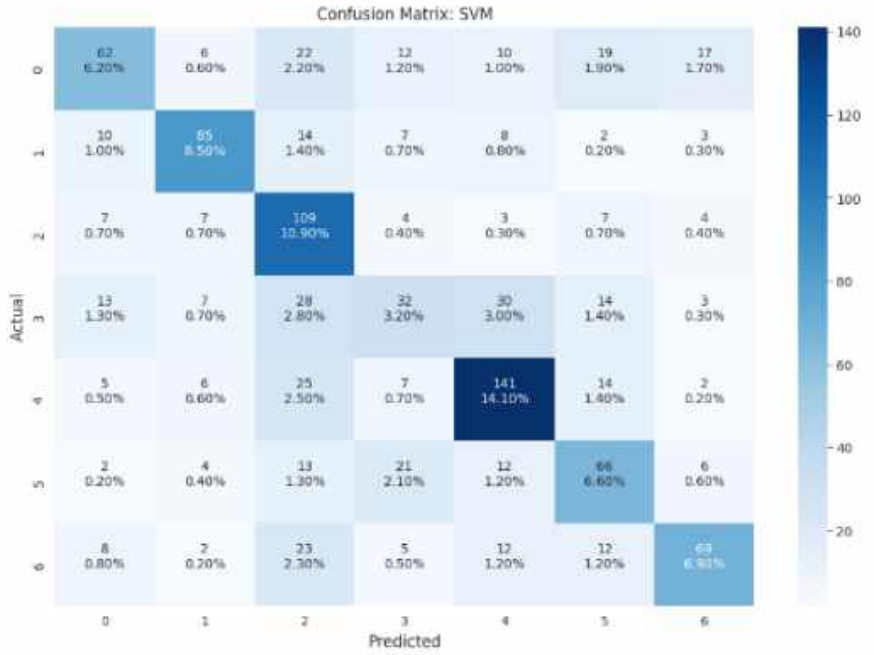



**Fig. 6.** Confusion matrix for SVM model

to its predominant theme or content which is as shown below:

- **Age:** Text related to age-based discrimination or stereotypes.
- **Gender:** Text containing gender-based biases or stereotypes.
- **Religion:** Text pertaining to religious affiliations or beliefs.
- **Mockery:** Text involving mocking or ridiculing individuals or groups.
- **Offensive:** Text with offensive language or content.
- **Abusive:** Text containing abusive or derogatory language.
- **Not-Cyberbullying:** Text not classified as cyberbullying based on the defined criteria.

Table 3 presents examples of sentences from each class category and Table 2 presents each class level data statistics.

## 4 Methodology

The following section presents a comprehensive overview of various techniques that are utilized for the prediction of cyberbullying on the MC-Hinglish$1.0$ dataset. We have used five machine learning algorithms, namely Support Vector Machines (SVM), Random Forest, Logistic Regression, Multinomial Naive Bayes, and XGBoost, to improve the prediction's accuracy. These algorithms are commonly employed in machine learning and have proven to be highly effective in earlier research.

Furthermore, we have applied an ensemble or hybrid model to combine five different machine learning models to enhance the model's performance (as shown in Fig. 3). By combining the predictions of several models, ensemble methods are known to improve the predictive ability of machine learning models. All things considered, putting these strategies to use can help reduce the harm that cyberbullying on online platforms causes.

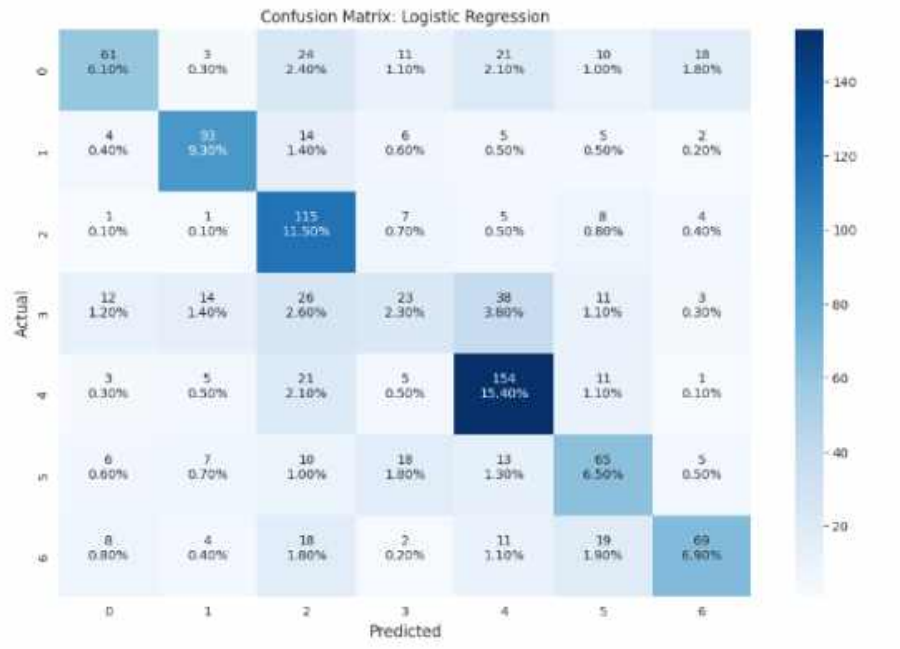



**Fig. 7.** Confusion matrix for logistic regression model

### 4.1 Machine Learning Models

**SVM:** This kind of binary linear classification model analyzes and classifies data by identifying patterns in it through the use of a learning algorithm. Finding decision boundaries and classifying a dataset are the primary goals of support vector machines (SVM) [7].

SVM is renowned for its accuracy even though its training time can be lengthy. SVM looks for the decision boundary that maximizes the difference in distance between the classes. SVM classifiers usually have the parameter values C=0.1 and kernel=rbf set.

Finding a hyperplane with the largest possible margin between the classes is the goal of the SVM equation. Support vectors, a subset of the training data that consists of the points that are closest to the decision boundary, are what determine the hyperplane. New data points can be accurately classified by SVM by optimizing the margin between the support vectors on either side of the hyperplane.

SVM is a learning algorithm-based binary linear classification model that is used to analyze, classify, and identify patterns in data [7]. SVM is used to identify decision boundaries and divide datasets into classes; despite its sometimes sluggish training period, SVM is an accurate algorithm:

$$\overrightarrow{W} = \sum_j \alpha_j c_j \overrightarrow{d}_j, \alpha_j \geq 0, \tag{1}$$

where the data's polarity (positive and negative) is represented by vector $\mathbf{w}$, which is a hyperplane $\in \{-1, 1\}$. By solving the dual optimization problem, $\alpha_j$ are obtained [7].

The only document vectors contributing are those whose value is greater than zero, and they are referred to as support vectors.

The process of classifying test instances involves identifying the side of the hyperplane on which they land.

**Logistic Regression:** It is a kind of regression model that is frequently applied to problems of

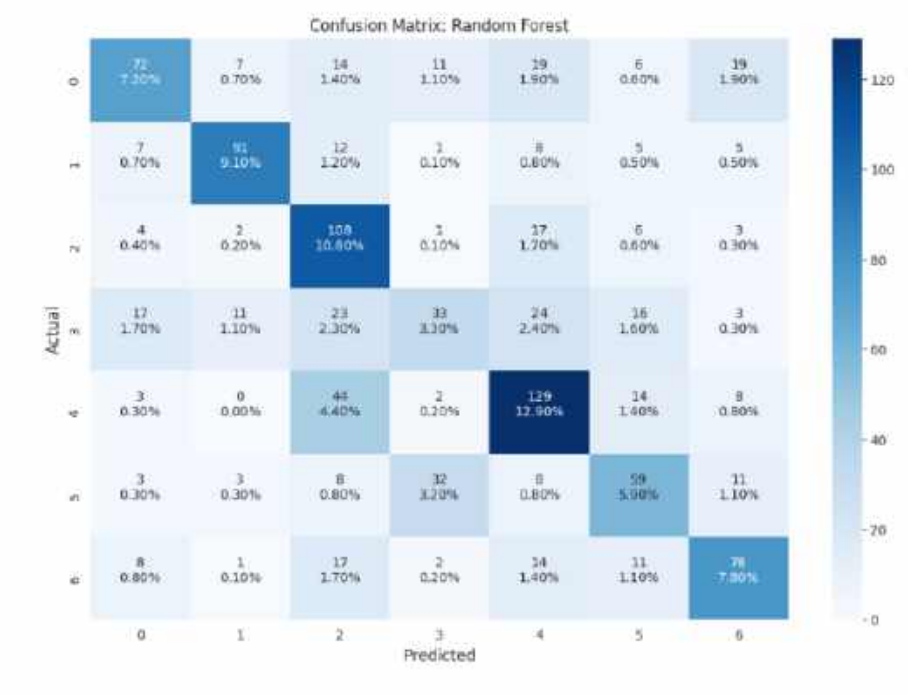



**Fig. 8.** Confusion matrix for random forest model

classification. It is especially helpful in scenarios where we have to forecast a binary result, meaning that the dependent variable can only have two possible values: 0 or 1. One or more independent variables and one categorical dependent variable are usually related using this model [4]. Selecting a hyper-plane that maximizes the gap between the classes' separation is the aim of logistic regression.

That is to say, it fits a line that is as accurate as possible in differentiating between the two classes. The recommended value for the LR classifiers' parameter values is $C = 3$, which is how this is accomplished [4]. This way, new data points are classified by the model in a way that allows it to predict an outcome's probability with accuracy. The output of the linear equation is converted into a probability value between 0 and 1 by the logistic regression model using a sigmoid function.

It is simpler to understand the output of the linear equation as a probability because the sigmoid function "squashes" it into a range between 0 and 1 [4]. Based on a threshold value, the model can categorize the data point into one of the two classes after obtaining the probability.

**Random Forest Classifier:** For tasks involving classification, this well-liked machine learning algorithm is frequently employed. The model's accuracy and robustness are increased by utilizing an ensemble approach that combines a number of decision tree classifiers [7]. The Random Forest classifier, to put it simply, uses the training dataset to construct several decision trees, then gathers votes from each tree to determine the final label or class of the test object.

To lessen the chance of overfitting to the training data, each decision tree in the forest is trained using a random subset of both the features and the training data [7]. With a parameter value of n estimators = 40, the Random Forest classifier has been tuned for this particular scenario.

This indicates that 40 decision trees, each with a random subset of the training data and features, are constructed by the classifier using the training set. The Random Forest classifier can produce a more reliable and accurate classification model

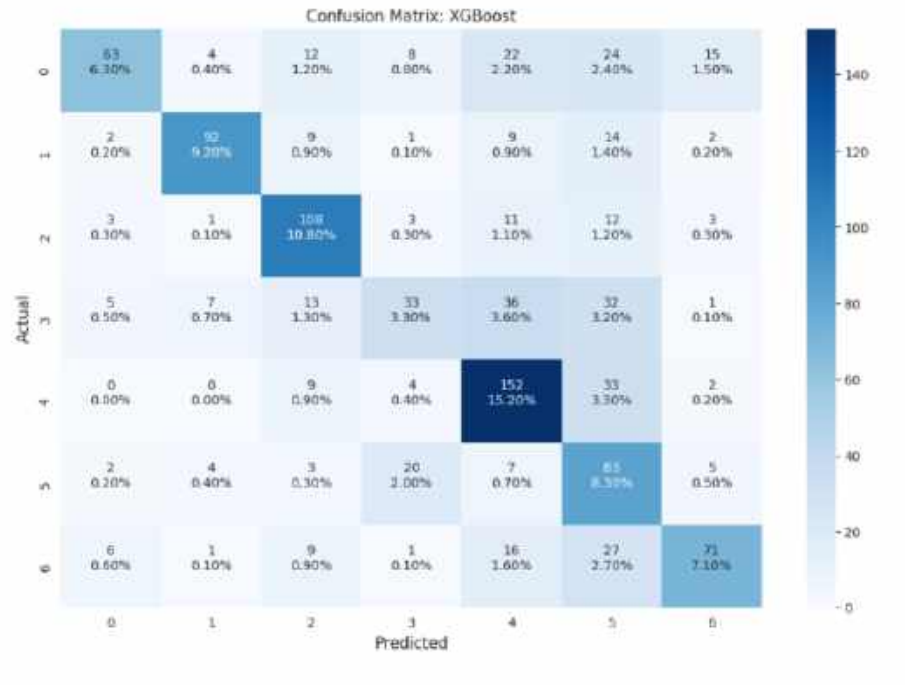



**Fig. 9.** Confusion matrix for XGBoost model

by integrating the predictions of these separate decision trees.

**Multinomial Naive Bayes:** MultinomialNB, part of machine learning's Naive Bayes family, is tailored for classification tasks involving multiple classes and features, assuming features' distributions follow a multinomial distribution. It computes probabilities for each class given input features using Bayes' theorem, making it particularly effective for text classification, such as sentiment analysis or document categorization [23]. Despite its "naive" assumption of feature independence, MultinomialNB often yields strong performance, especially with high-dimensional data like word counts. Its simplicity, efficiency, and ability to handle sparse data make it a popular choice for many text-based machine learning applications.

**XGBoost Classifier:** The XGBoost (Extreme Gradient Boosting) classifier [14] is a powerful machine learning algorithm known for its effectiveness in classification tasks. With the provided parameters—n-estimators set to 500, max-depth at 6, and learning-rate set to 0.1—the classifier is configured to create a strong ensemble of decision trees.

The parameter n-estimators determine the number of boosting rounds or trees to be generated, in this case, a substantial 500, potentially enhancing the model's predictive capacity. Setting max-depth to 6 controls the maximum depth of each decision tree, preventing overfitting while allowing for sufficient complexity to capture patterns in the data.

A learning-rate of 0.1 dictates the step size during optimization, influencing the rate at which the algorithm adapts to minimize the loss function. This parameter choice is often balanced, as it's neither too aggressive nor too conservative, facilitating a stable learning process [23].

Overall, this configuration aims to strike a balance between model complexity, computational efficiency, and generalization performance, making it a robust choice for classification tasks across various domains.

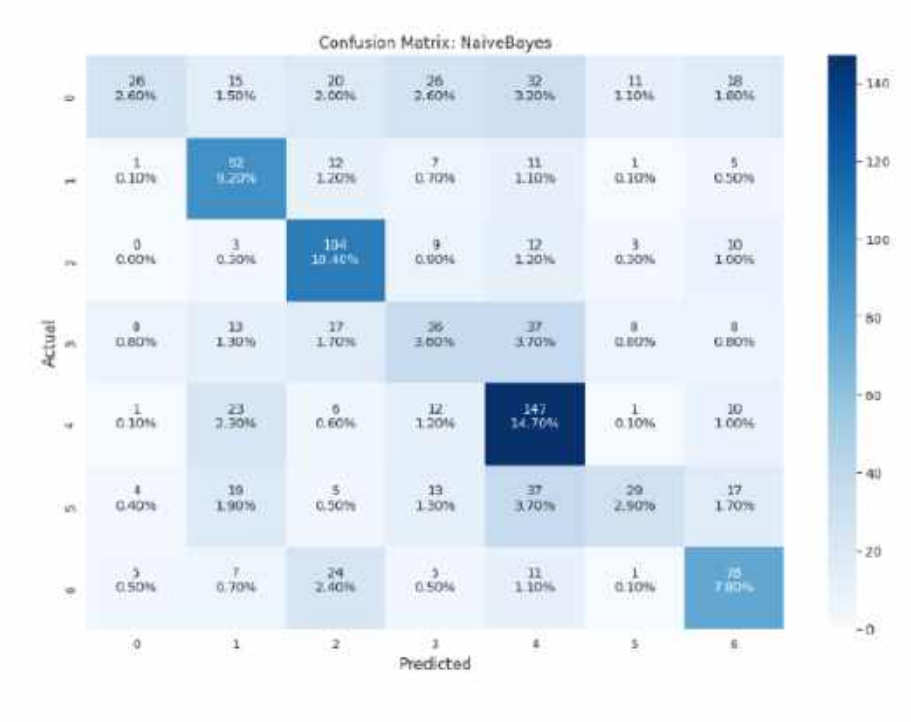



**Fig. 10.** Confusion matrix for naive bayes model

**Ensemble Learning Model:** The ensemble learning [6] is used to raise baseline learning techniques' accuracy and performance. SVM, Random Forest, Logistic Regression, Multinomial Naive Bayes, and XGBoost are the base learners utilized in this methodology.

For every base classifier in the ensemble classifier, the labels for age, gender, religion, mockery, abusive behavior, offensiveness, and not-cyberbullying are determined.

Ultimately, by combining the predictions from all of the classifiers to determine the outcome, a soft voting mechanism allows these classifiers to work together to create an ensemble model.

The final prediction is determined by tallying the votes cast by the base learners for the class label (as shown in Fig. 4).

## 5 Result and Analysis

### 5.1 Experimental Result

We evaluated the performance of five machine learning models - Multinomial Naive Bayes (MNB), Support Vector Machines (SVM), Random Forest (RF), XGBoost, and Logistic Regression (LR) and then applied an ensemble learning model to enhance the accuracy of cyberbullying detection.

The ensemble learning model has been built using five machine learning models, namely SVM, Multinomial Naive Bayes, XGBoost, Logistic Regression, and Random Forest. To assess the model's performance, we have considered standard evaluation metrics like accuracy, recall, and F1 scores. The experimental results are presented in Table 4. Accuracy measures the proportion of correctly classified instances out of all instances in a given dataset [6]. The text goes on to mention that the mean accuracy across all models is approximately 60.90%.

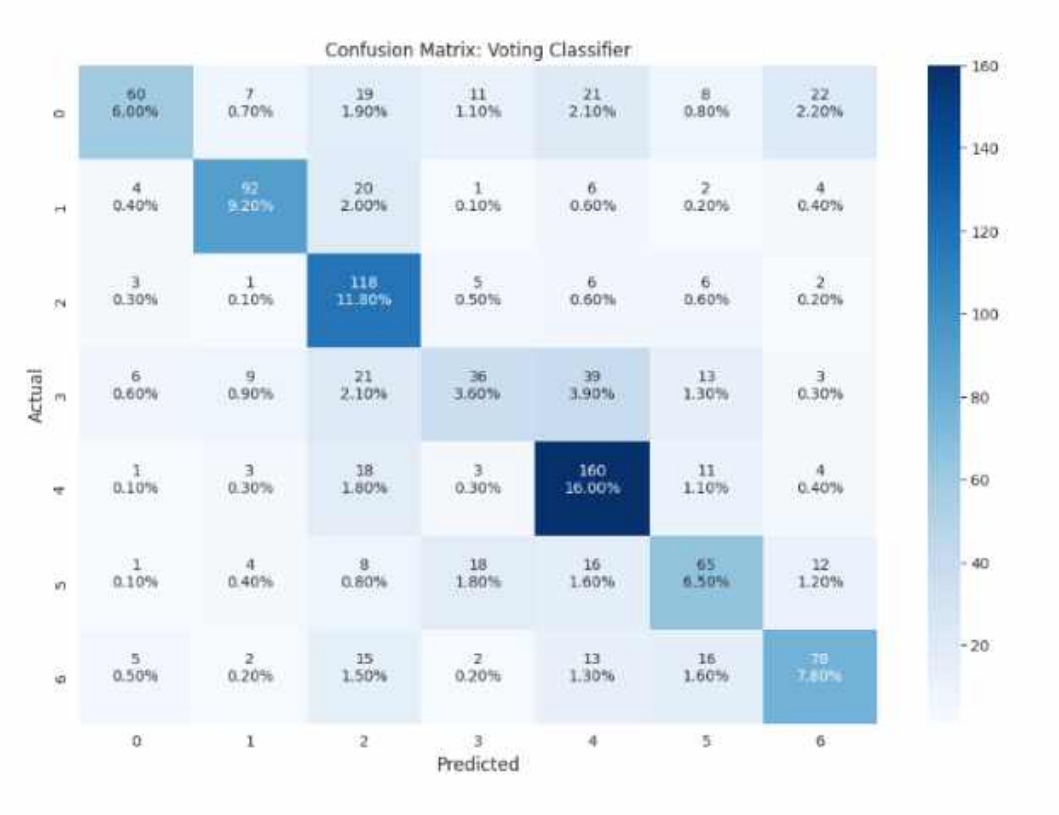



**Fig. 11.** Confusion matrix for ensemble model (voting classifier)

This means that the model is able to correctly classify around 61.00% of the instances on average. The recall metric, also referred to as sensitivity, assesses the accuracy of the model in identifying true positive instances out of all the actual positive instances [6]. A high recall value indicates that the model is successful in identifying a majority of positive instances.

On average, the recall score stands at approximately 59.40%, which means that the model accurately detects around 59.00% of the actual positive instances. The F1-score is a calculation that combines precision and recall in a balanced way. It provides a fair evaluation of the model's performance by considering both precision and recall. The average F1-score is around 58.79%, which shows that the model's precision and recall are reasonably balanced.

### 5.2 Analysis

A confusion matrix is a table that is used to evaluate the performance of a classification model on a set of test data for which the true values are known. It shows the number of correct and incorrect predictions made by the classification model, broken down by each class.

The confusion matrix is typically represented as an $N * N$ matrix, where N is the number of classes in the classification problem. Each row in the matrix represents the instances of an actual class, while each column represents the instances of a predicted class [26]. The main entries in the confusion matrix are:

1. **True Positives (TP):** The number of instances that were correctly predicted as belonging to the positive class.

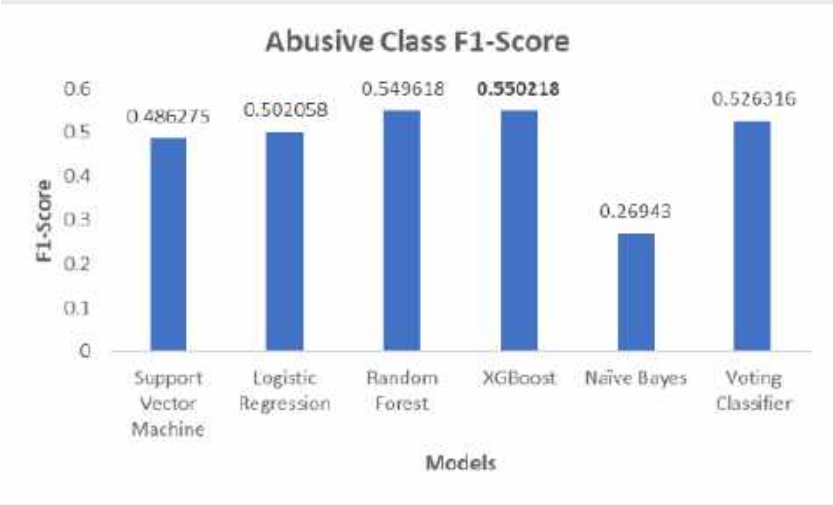



**Fig. 12.** Comparison of F1-Scores for abusive level classification across ML models

2. **True Negatives (TN):** The number of instances that were correctly predicted as belonging to the negative class.
3. **False Positives (FP):** The number of instances that were incorrectly predicted as belonging to the positive class.
4. **False Negatives (FN):** The number of instances that were incorrectly predicted as belonging to the negative class.

The values in the confusion matrix can be used to calculate various performance metrics for the classification model, such as accuracy, precision, recall, and F1-score [26]. The confusion matrix provides a comprehensive view of the model's performance, allowing for a better understanding of its strengths and weaknesses, and can guide further improvements or adjustments to the model.

The results showcase the performance of various classification models from the confusion matrix on a cyberbullying Twitter dataset with six classes: abusive, age, gender, mockery, not-cyberbullying, offensive, and religion. The Multinomial Naive Bayes (MNB) and Voting models have complete information for calculating overall accuracy and class-wise performance. The Voting model outperforms MNB in terms of overall accuracy, achieving 60.9% compared to 51.2% for MNB. However, upon closer inspection, the models exhibit varying strengths and weaknesses across different classes. For the abusive class, the Voting model correctly identified 80 instances, significantly higher than MNB's 45 true positives, indicating better performance in detecting abusive content.

Conversely, MNB excelled in identifying age-related instances (172 true positives) and mockery cases (108 true positives) compared to the Voting model's 118 and 76 true positives, respectively.

Interestingly, the Voting model demonstrated superior performance in recognizing gender-related instances (219 true positives) and offensive content (121 true positives), while MNB performed better in identifying non-cyberbullying instances (287 true positives) and religion-related cases (146 true positives).

From the confusion matrix, we can observe that the Voting Classifier performed well in identifying instances of "Age," "Gender," "Religion," "Not-cyberbullying," and "Offensive" classes. However, it had some difficulty in distinguishing between "Mockery" and "Abusive" classes, as evident from the off-diagonal entries.

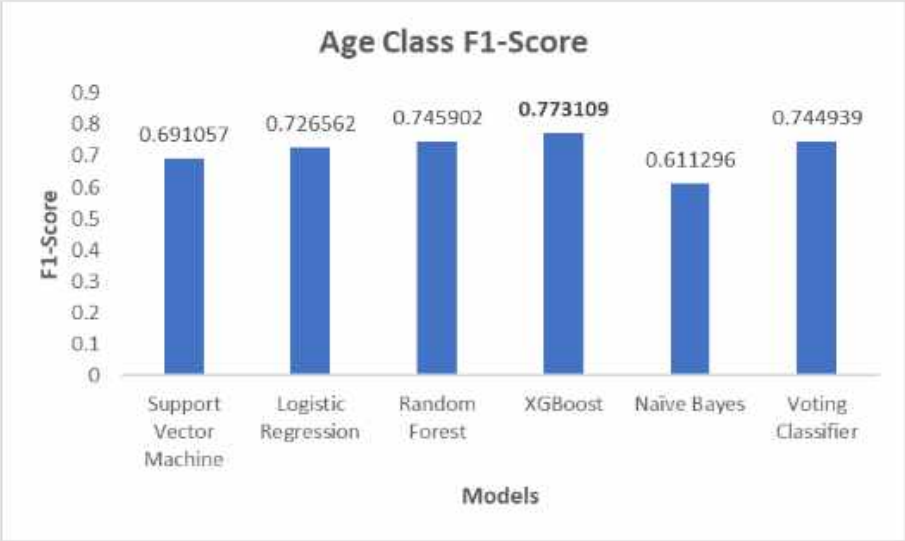



**Fig. 13.** Comparison of F1-Scores for age level classification across ML models

### 5.3 Individual Class Level Analysis

The graph shows the F1-scores for classifying abusive content using different models. The XGBoost model achieves the highest F1-score of around 0.55, it means that among all the models compared in the graph, the XGBoost model performed the best in correctly identifying and classifying abusive content.

An F1-score of 0.55 indicates that the XGBoost model strikes a good balance between correctly detecting true positives (abusive content identified as such) and avoiding false positives (non-abusive content misclassified as abusive). The Voting Classifier and Random Forest models have the next highest scores around 0.526. The Support Vector Machine model has the lowest F1-score of approximately 0.486.

The graph shows the F1-scores for classifying age content using different models. The XGBoost model achieves the highest F1-score of around 0.77, it means that among all the models compared in the graph, the XGBoost model performed the best in correctly identifying and classifying age content. An F1-score of 0.77 indicates that the XGBoost model strikes a good balance between correctly detecting true positives (age content identified as such) and avoiding false positives (non-age content misclassified as age). The Voting Classifier and Random Forest models have the next highest scores around 0.74. The Naïve Bayes model has the lowest F1-score of approximately 0.611.

The graph shows the F1-scores for classifying gender content using different models. The XGBoost model achieves the highest F1-score of around 0.71, it means that among all the models compared in the graph, the XGBoost model performed the best in correctly identifying and classifying gender content.

An F1-score of 0.71 indicates that the XGBoost model strikes a good balance between correctly detecting true positives (gender content identified as such) and avoiding false positives (non-gender content misclassified as gender). The Support Vector Machine model has the lowest F1-score of approximately 0.581.

The graph shows the F1-scores for classifying mockery content using different models. The Voting Classifier model achieves the highest F1-score of around 0.35, it means that among all the models compared in the graph, the Voting Classifier model performed the best in correctly identifying and classifying mockery content.

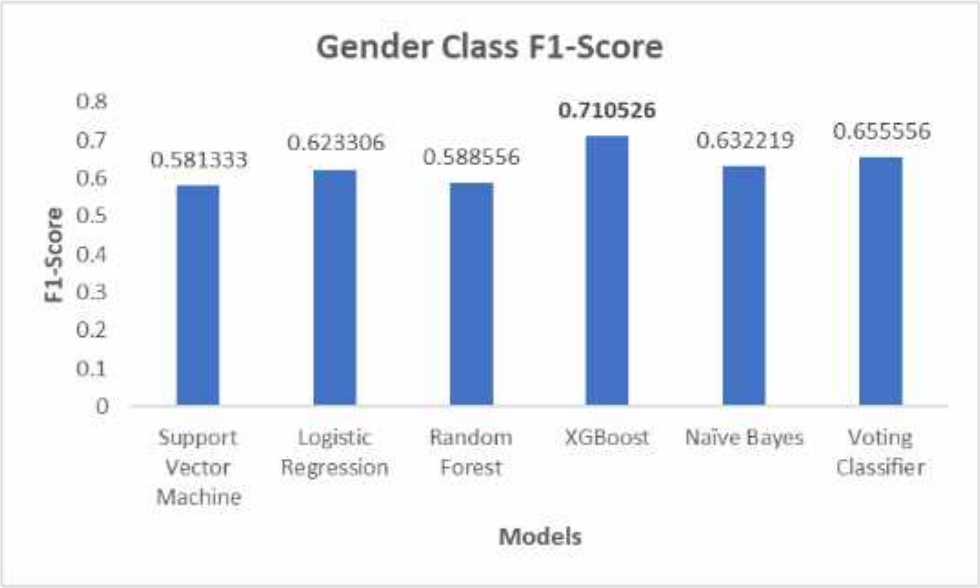



**Fig. 14.** Comparison of F1-Scores for gender content classification across ML models

An F1-score of 0.35 indicates that the Voting Classifier model strikes a good balance between correctly detecting true positives (mockery content identified as such) and avoiding false positives (non-mockery content misclassified as mockery). The Logistic Regression model has the lowest F1-score of approximately 0.231.

The graph shows the F1-scores for classifying not-cyberbullying content using different models. The Voting Classifier model achieves the highest F1-score of around 0.69, it means that among all the models compared in the graph, the Voting Classifier model performed the best in correctly identifying and classifying not-cyberbullying content.

An F1-score of 0.69 indicates that the Voting Classifier model strikes a good balance between correctly detecting true positives (not-cyberbullying content identified as such) and avoiding false positives (non- not-cyberbullying content misclassified as not-cyberbullying). The Naïve Bayes model has the lowest F1-score of approximately 0.603. The graph shows the F1-scores for classifying offensive content using different models.

The Voting Classifier model achieves the highest F1-score of around 0.53, it means that among all the models compared in the graph, the Voting Classifier model performed the best in correctly identifying and classifying offensive content. An F1-score of 0.53 indicates that the Voting Classifier model strikes a good balance between correctly detecting true positives (offensive content identified as such) and avoiding false positives (non-offensive content misclassified as offensive).

The Naïve Bayes model has the lowest F1-score of approximately 0.325. The graph shows the F1-scores for classifying religion content using different models. The XGBoost model achieves the highest F1-score of around 0.617, it means that among all the models compared in the graph, the XGBoost model performed the best in correctly identifying and classifying religion content.

An F1-score of 0.617 indicates that the XGBoost model strikes a good balance between correctly detecting true positives (religion content identified as such) and avoiding false positives (non-religion content misclassified as religion). The Naïve Bayes model has the lowest F1-score of approximately 0.563.

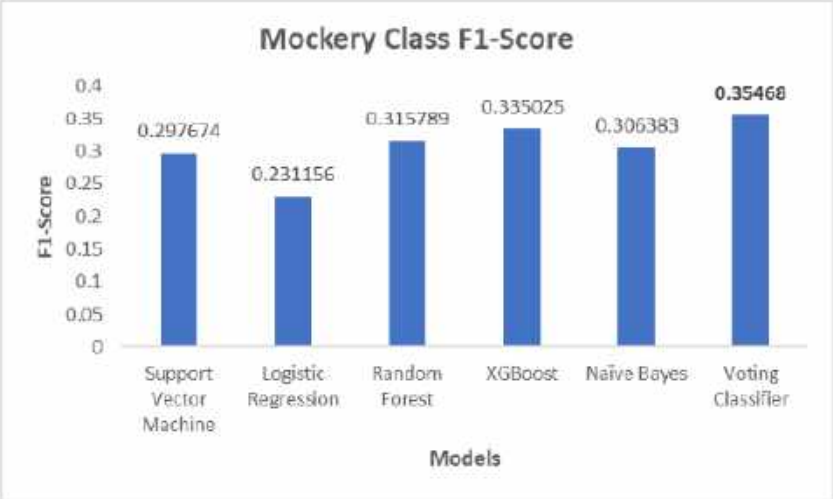



**Fig. 15.** Comparison of F1-Scores for mockery content classification across ML models

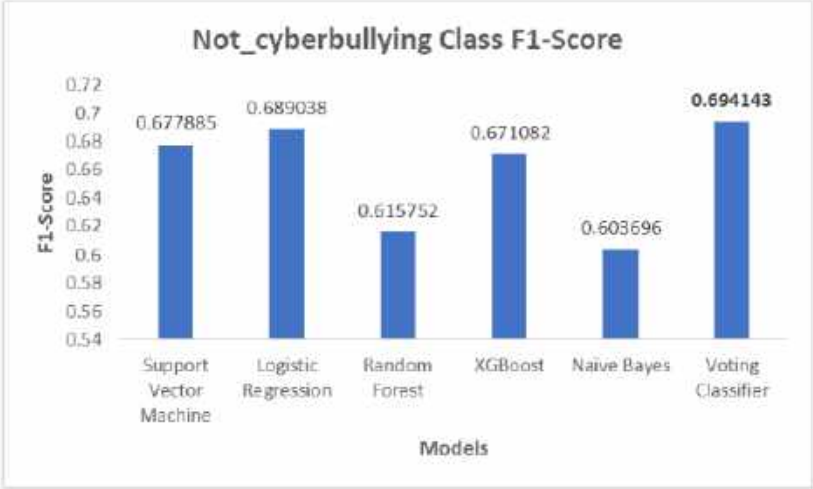



**Fig. 16.** Comparison of F1-Scores for not-cyberbullying content classification across ML models

## 6 Conclusion and Future Work

In this paper, we have presented a novel multi-class dataset, MC-Hinglish1.0 consisting of 8,400 Hinglish comments, annotated across seven cyberbullying categories: age, gender, religion, mockery, abusive, offensive, and not cyberbullying. We have evaluated the performance of different machine learning models, namely, Multinomial Naive Bayes, Support Vector Machines, Random Forest, XGBoost, and Logistic Regression – along with an ensemble model (voting classifier). Our experiments demonstrated the effectiveness of these models, with the XGBoost model achieving the highest individual performance with an accuracy of 0.602 and a weighted F1-score of 0.5804. However, the ensemble voting classifier outperformed all individual models, obtaining an accuracy of 0.609 and a weighted F1-score of 0.5879.

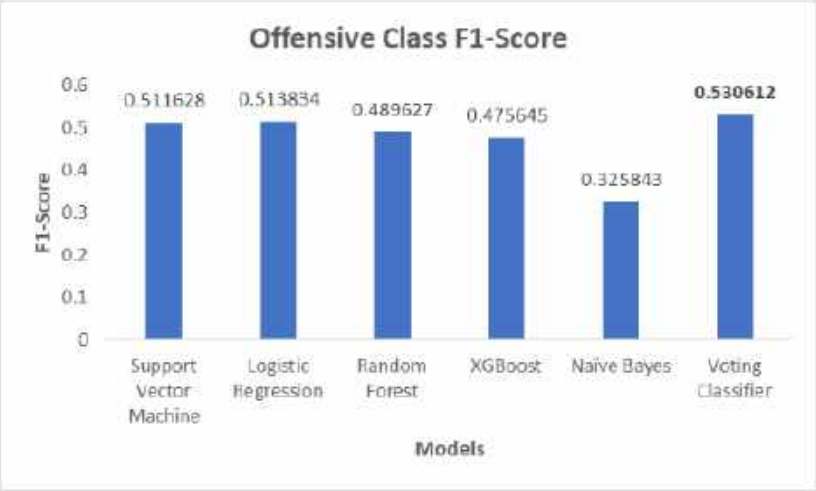



**Fig. 17.** Comparison of F1-Scores for offensive content classification across ML models

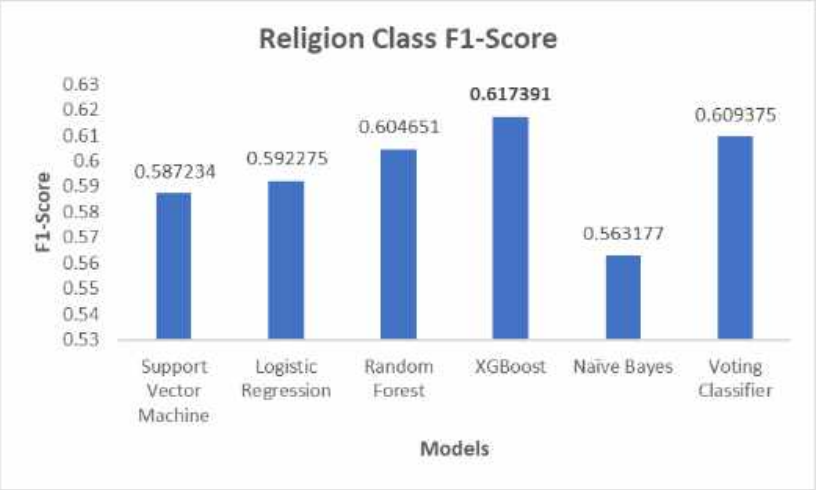



**Fig. 18.** Comparison of F1-Scores for religion content classification across ML models

The confusion matrix analysis revealed that while the Voting Classifier excelled in identifying instances of ”Age,” ”Gender,” ”Religion,” ”Offensive,” and ”Not Cyberbullying” classes, it faced challenges in distinguishing between certain ”Mockery,” and ”Abusive” classes. Our study highlights the importance of developing robust cyberbullying detection models tailored for code-mixed languages like Hinglish. The insights gained from this research can contribute to creating safer online spaces and promoting inclusivity in the digital realm. The future scopes of this work are summarized as follows:

– **Dataset Expansion:** As the need for robust cyberbullying detection systems grows, future work will involve expanding the dataset size. This includes acquiring additional data from diverse sources and languages to enhance the generalization capabilities of the model [15].

In addition to the Hinglish dataset, preparation of other multi-classification code-mixed Indic datasets, such as Binglish, will be undertaken. This expansion aims to capture the nuances and variations present in different languages and language mixes commonly used in online communication.

– **Investigation of Efficient Deep Learning-based Approaches:** To further improve cyberbullying detection accuracy and efficiency, future work will focus on exploring efficient deep learning-based approaches. This involves leveraging advancements in deep learning architectures, such as transformers, convolutional neural networks (CNNs), and recurrent neural networks (RNNs), tailored specifically for code-mixed text data [15].

  By designing models that effectively capture the complex linguistic patterns present in code-mixed text, the performance of cyberbullying detection systems can be significantly enhanced.

– **Incorporation of Multimodal Data:** Incorporating multimodal data, such as text, images, and user metadata, into the cyberbullying detection framework presents another avenue for future research [15].

  By analyzing multiple modalities simultaneously, the model can gain a more comprehensive understanding of online interactions, thereby improving the detection accuracy and robustness. Fusion techniques that combine information from different modalities will be explored to leverage the complementary nature of multimodal data [7].

– **Fine-tuning and Transfer Learning:** Fine-tuning pre-trained language models on cyberbullying detection tasks and leveraging transfer learning techniques will be investigated as a means to improve model performance. By leveraging large-scale pre-trained models such as BERT, GPT, or XLNet, the model can benefit from learning representations of text data across multiple languages and domains [7]. Fine-tuning these models on the specific cyberbullying detection task can help capture domain-specific features and improve overall performance.

– **Deployment and Real-world Application:** Finally, future work will involve the deployment and real-world application of the developed cyberbullying detection system. This includes integrating the model into social media platforms, online forums, and other digital communication channels to proactively identify and mitigate instances of cyberbullying [7].

  Continuous monitoring and refinement of the model based on user feedback and evolving linguistic trends will be essential to ensure its effectiveness in real-world scenarios.

## References

1. **Ahammed, S., Rahman, M., Niloy, M. H., Chowdhury, S. M. M. H. (2019).** Implementation of machine learning to detect hate speech in bangla language. Proceedings of the 8th International Conference System Modeling and Advancement in Research Trends, pp. 317–320. DOI: 10.1109/SMART46866.2019.9117214.

2. **Ahmed, M. T., Rahman, M., Nur, S., Islam, A., Das, D. (2021).** Deployment of machine learning and deep learning algorithms in detecting cyberbullying in bangla and romanized bangla text: A comparative study. Proceedings of the International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies, pp. 1–10. DOI: 10.1109/ICAECT49130.2021.9392608.

3. **Akhter, A., Uzzal, K. A., Polash, M. (2019).** Cyber bullying detection and classification using multinomial Naïve Bayes and fuzzy logic. International Journal of Mathematical Sciences and Computing, Vol. 5, No. 4, pp. 1–12. DOI: 10.5815/ijmsc.2019.04.01.

4. **Alam, K., Bhowmik, S., Prosun, P. (2021).** Cyberbullying detection: An ensemble based machine learning approach. Proceedings of the third international conference on intelligent communication technologies

and virtual mobile networks, pp. 710–715. DOI: 10.1109/ICICV50876.2021.9388499.

5. **Ali, M. U., Lefticaru, R. (2024).** Detection of cyberbullying on social media platforms using machine learning. UK Workshop on Computational Intelligence, Springer Nature Switzerland, pp. 220–233. DOI: 10.1007/978-3-031-47508-5_18.
6. **Alqahtani, A. F., Ilyas, M. (2024).** An ensemble-based multi-classification machine learning classifiers approach to detect multiple classes of cyberbullying. Machine Learning and Knowledge Extraction, Vol. 6, No. 1, pp. 156–170. DOI: 10.3390/make6010009.
7. **Alqahtani, A. F., Ilyas, M. (2024).** A machine learning ensemble model for the detection of cyberbullying. DOI: 10.48550/arXiv.2402.12538e.
8. **Atoum, J. O. (2020).** Cyberbullying detection through sentiment analysis. Proceedings of the International Conference on Computational Science and Computational Intelligence, pp. 292–297. DOI: 10.1109/CSCI51800.2020.00056.
9. **Balakrishnan, V., Khan, S., Arabnia, H. R. (2020).** Improving cyberbullying detection using twitter users' psychological features and machine learning. Computers & Security, Vol. 90, pp. 101710. DOI: 10.1016/j.cose.2019.101710.
10. **Bhalla, A., Chadha, A. (2023).** An efficient model to detect the presence of hinglish text in youtube data. Proceedings of the International Conference on Advances in Computation, Communication and Information Technology, pp. 385–391. DOI: 10.1109/ICAICCIT60255.2023.10465821.
11. **Chakraborty, P., Seddiqui, M. H. (2019).** Threat and abusive language detection on social media in bengali language. Proceedings of the 1st International Conference on Advances in Science, Engineering and Robotics Technology, pp. 1–6. DOI: 10.1109/ICASERT.2019.8934609.
12. **Dubey, N., Kaushal, R. (2023).** Towards detection of cyberbullying in hinglish code mixed data. Proceedings of the 10th International Conference on Computing for Sustainable Global Development, pp. 1096–1100.
13. **Ghosh, R., Nowal, S., Manju, G. (2021).** Social media cyberbullying detection using machine learning in bengali language. International Journal of Engineering Research & Technology, Vol. 10, No. 5.
14. **Ji, C., Zou, X., Hu, Y., Liu, S., Lyu, L., Zheng, X. (2018).** XG-SF: an XGBoost classifier based on shapelet features for time series classification. 2018 International Conference on Identification, Information and Knowledge in the Internet of Things, Elsevier, Vol. 147, pp. 24–28. DOI: 10.1016/J.PROCS.2019.01.179.
15. **Kumar, S., Mondal, M., Dutta, T., Singh, T. D. (2024).** Cyberbullying detection in hinglish comments from social media using machine learning techniques. Multimedia Tools and Applications, pp. 1–22.
16. **Maity, K., Jain, R., Jha, P., Saha, S., Bhattacharyya, P. (2023).** Genex: A commonsense-aware unified generative framework for explainable cyberbullying detection. Conference on Empirical Methods in Natural Language Processing, pp. 16632–16645. DOI: 10.18653/v1/2023.emnlp-main.1035.
17. **Maity, K., Jha, P., Jain, R., Saha, S., Bhattacharyya, P. (2024).** "Explain thyself bully": Sentiment aided cyberbullying detection with explanation. Proceedings of the IEEE International Conference on Document Analysis and Recognition, pp. 132–148. DOI: 10.1007/978-3-031-41682-8_9.
18. **Maity, K., Kumar, A., Saha, S. (2022).** A multitask multimodal framework for sentiment and emotion-aided cyberbullying detection. IEEE Internet Computing, Vol. 26, No. 4, pp. 68–78. DOI: 10.1109/MIC.2022.3158583.

19. **Maity, K., Saha, S., Bhattacharyya, P. (2023).** Emoji, sentiment and emotion aided cyberbullying detection in hinglish. IEEE Transactions on Computational Social Systems, Vol. 10, No. 5, pp. 2411–2420. DOI: 10.1109/TCSS.2022.3183046.

20. **Mladenovic, M., Osmjanski, V., Vujicic-Stankovic, S. (2022).** Cyber-aggression, cyberbullying, and cyber-grooming: A survey and research challenges. ACM Computing Surveys, Vol. 54, No. 1, pp. 1–42. DOI: 10.1145/3424246.

21. **Muneer, A., Fati, S. M. (2020).** A comparative analysis of machine learning techniques for cyberbullying detection on twitter. Future Internet, Vol. 12, No. 4, pp. 187. DOI: 10.3390/fi12110187.

22. **Nath, S. S., Karim, R., Miraz, M. H. (2024).** Deep learning based cyberbullying detection in bangla language. arXiv. DOI: 10.48550/ARXIV.2401.06787.

23. **Nuthalapati, P., Abbaraju, S. A., Varma, G. H., Biswas, S. (2024).** Cyberbullying detection: A comparative study of classification algorithms. International Journal of Computer Science and Mobile Computing. DOI: 10.22541/au.170664263.38254624/v1.

24. **Pawar, R., Raje, R. R. (2019).** Multilingual cyberbullying detection system. Proceedings of the IEEE International Conference on Electro Information Technology (EIT), pp. 040–044. DOI: 10.1109/EIT.2019.8833846.

25. **Salawu, S., He, Y., Lumsden, J. (2020).** Approaches to automated detection of cyberbullying: A survey. IEEE Transactions on Affective Computing, Vol. 11, No. 1, pp. 3–24. DOI: 10.1109/TAFFC.2017.2761757.

26. **Singh, N. M., Sharma, S. K. (2023).** An efficient automated multi-modal cyberbullying detection using decision fusion classifier on social media platforms. Multimedia Tools and Applications, Vol. 83, No. 7, pp. 20507–20535. DOI: 10.1007/s11042-024-19031-z.

27. **Tripto, N. I., Eunus-Ali, M. (2018).** Detecting multilabel sentiment and emotions from bangla youtube comments. Proceedings of the International Conference on Bangla Speech and Language Processing (ICBSLP), pp. 1–6. DOI: 10.1109/ICBSLP.2018.8554875.

28. **Van-Hee, C., Jacobs, G., Emmery, C., Desmet, B., Lefever, E., Verhoeven, B., De-Pauw, G., Daelemans, W., Hoste, V. (2018).** Automatic detection of cyberbullying in social media text. PloS ONE, Vol. 13, No. 10, pp. 1–22. DOI: 10.1371/journal.pone.0203794.

29. **Verma, K., Milosevic, T., Cortis, K., Davis, B. (2022).** Benchmarking language models for cyberbullying identification and classification from social-media texts. Proceedings of the First Workshop on Language Technology and Resources for a Fair, Inclusive, and Safe Society within the 13th Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, pp. 26–31.

*Article received on 22/05/2024; accepted on 06/06/2024.*
**Corresponding author is Sahinur Rahman Laskar.*

ISSN 2007-9737

# Creation of a Corpus in Spanish for the Recognition of Personality Traits

Víctor Manuel Bátiz-Beltrán, María Lucía Barrón-Estrada*,
Ramón Zatarain-Cabada, Jonathan Iván Roldán-Arana

Tecnológico Nacional de México Campus Culiacán,
Posgrado e Investigación, Culiacán, Sinaloa,
Mexico

{victor.bb, lucia.be, ramon.zc}@culiacan.tecnm.mx, jonathan_roldan@itculiacan.edu.mx

**Abstract.** Automatic personality recognition is a research area that has become very important in recent years. Currently there is research with different approaches that seek to automatically recognize personality traits by means of text. There are methods that use texts from voice transcriptions or texts written by people to determine whether an individual has a certain personality trait. These methods are based on machine learning and deep learning algorithms. A key element for the construction of such models is to have a data set (corpus) for training and optimization. This paper presents the creation of a corpus called PersonText, which contains 213 texts in Spanish with their respective labels related to the presence or absence of the personality traits of the Big-Five model, as well as the scores obtained by the participants in a standardized personality test. The main motivation for the creation of this corpus was the limited existence of corpora of texts in Spanish focused on personality recognition. The information was obtained from the platform called PersonApp, used for data collection based on standardized personality tests and videos of the participants. Additionally, to evaluate the corpus, tests were performed with different machine learning and deep learning models. The results obtained are promising and validate the relevance of the corpus built to address the task of automatic personality recognition.

**Keywords.** Big-Five, corpus, personality, machine learning, deep learning, machine recognition, machine learning, machine recognition.

## 1 Introduction

Personality is a set of traits that determine the way a person acts or makes decisions. These traits come to be aspects such as emotions, feelings and pattern characteristics of attitudes and thoughts [1]. Personality is what induces us to make decisions, such as choosing the type of clothes we wear every day, the items we buy at the supermarket, the hobbies we have, the places we visit or even the social events we attend. Therefore, automatic recognition of personality traits is important because it allows us to solve problems more efficiently, such as placing the right person in the right workplace or improving educational processes for students according to their personality.

Currently, two of the most common and widely used models for personality recognition based on written tests are the Myers-Briggs Type Indicator (MBTI) model, which is a self-reported questionnaire that aims to assess how individuals perceive their surroundings and make decisions. [2] and Big-Five personality traits [3] (also called OCEAN)

The MBTI test is not accepted in scientific circles due to a lack of evidence demonstrating its reliability as a personality test. In the other hand, the Big-Five model is the most popular and accepted model by the scientific community and is the one used as the basis for this research. In this model the five major personality traits are represented by Openness to Experience, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.

To automatically recognize the personality of an individual, machine learning (ML) or deep learning (DL) methods are used, which in turn require data sets or text corpora for training the classifiers that will automatically determine the presence or absence of each personality trait. There are

corpora in languages other than Spanish for personality recognition, as is the case of the myPersonality corpus, which was the result of a project based on a Facebook application that allowed participants to take a personality test and give their consent for the program to record their profile information [4].

Another corpus with data based on personality traits is the b5 corpus [5], which is a collection of controlled and free texts (without specific subject matter) produced in different communicative tasks (e.g., referential, or descriptive), and accompanied by personality inventories of their authors and other demographic data.

When approaching the task of automatic personality recognition based on Spanish texts, one of the problems found is the low number of corpora based on Spanish texts oriented to personality recognition.

In this research work the following research question is addressed:

– Is it possible to build ML and DL models for automatic personality trait recognition that obtain a performance comparable to the state of the art when applied to a Spanish corpus constructed from textual transcripts of Spanish videos and their corresponding personality trait labels based on a standardized personality test?

The main contribution of this work is the creation of a corpus of texts in Spanish language, which is based on the Big-Five model for personality traits using a novel methodology for its construction, as well as the evaluation of the corpus by means of classification models by ML and DL methods.

The data were obtained from PersonApp [6], a platform developed internally by our research group that aims to collect data through videos and standardized personality tests of the participants, based on the questions available in the International Personality Item Pool (IPIP) [7]. In addition, several classification models were developed and optimized to test the corpus.

This paper is structured in the following order: in Section 2 we present related work in text-based personality recognition; Section 3 presents the methodology implemented for this study; Section 4 shows the results with the classification models and their discussion; finally, Section 5 describes the conclusions and future work.

## 2 Related Work

This section describes related work in the field of personality recognition using text. The section is divided into two parts: the first part presents works on creating a text dataset (corpus) based on personality recognition models. The second part describes works on automatic recognition where datasets are used for training and classification of personality in text.

### 2.1 Text Corpora About Personality

There are several corpora used for text-based personality prediction. One of the first such corpus is called essays [8], created by Pennebaker and King, which is a dataset containing approximately 2400 stream-of-consciousness essays, labeled with the personality traits of the authors of each piece of writing.

The corpus is organized as follows:

– Author identifier.
– Text of the authors' stream-of-consciousness essays.
– Personality traits labeled in two classes (yes and no, to indicate presence or absence of each of the Big-Five traits). The field titles for the personality traits are EXT (Extraversion), NEU (Neuroticism), AGR (Agreeableness), CON (Conscientiousness) and OPN (Openness to Experience). The questionnaire used to collect personality traits was the Five-Factor Inventory [9].

In the work of Kim & Walter [10], the corpus personae used for the prediction of authorship attribution and author personality is presented. The corpus consists of essays written by 145 authors (approximately 1,400 words each), as well as personality profiles based on the MBTI test.

This test employs four scales, defined as opposing pairs between eight categories: introversion-extraversion, sensing-intuition, thinking-feeling, and judgment-perception.

Each category is symbolized by a letter, and the result is expressed as a combination of four letters,

**Table 1.** Text corpora based on Big Five model

| Corpus | Size (number of records) | Language | Labels | Measured values of personality traits |
|---|---|---|---|---|
| myPersonality subcorpus [13] | 10,000 | English | Big-Five (y/n) | 0 to 5 |
| PAN-AP-2015 [11] | 726 | English, Spanish, Italian and Dutch | Big-Five (numeric) | -0.5 to +0.5 |
| b5-post [5] | 194,382 | Portuguese | Big-Five (yes/no) | - |
| Essays [8] | 2,400 | English | Big-Five (y/n) | - |
| HWxPI [14] | 836 | Spanish | Big-Five (numeric) | 0 or 1 |

among the sixteen possible, which is intended to define the personality of the subject [2].

There are also corpora whose data were collected from the Facebook application. Among these we can mention the b5 corpus which is a corpus in Portuguese language. The b5 corpus is composed of several subcorpora, being the b5-post subcorpus the largest, since it has 194,382 sentences from 1,019 users [5]. The subcorpus was constructed for the purpose of conducting research and developing computational models for the recognition of personality traits and the profiling of authors.

In [11], Rangel et al. mention the PAN-AP-2015 corpus which is composed of texts collected from the social network Twitter (now known as X), from 726 users, in four languages: English, Spanish, Italian and Dutch. The texts are distributed as follows: 336 are in English, 228 are in Spanish, 86 are in Italian and 76 are in Dutch. The assessment of personality traits was conducted via the BFI-10 online test [12], which was completed by the participants themselves. The corpus is annotated with gender and personality traits in normalized values between -0.5 and +0.5.

Similarly, myPersonality is one of the most popular and widely used corpora for text-based personality recognition tests. This corpus is a database that was constructed by collecting information from Facebook, and where the data are labeled similarly to the essays dataset, with binary values (y/n) to indicate whether the trait is present or not, as well as with their respective personality measurement values obtained by IPIP tests [7] corresponding to each author. In order to assess the performance of the automatic recognition models presented in this research, a subset of the myPersonality corpus, as proposed in [13] by Celli et al., was utilized.

In the Spanish language in [14], the authors present the HWxPI corpus consisting of handwritten essays for personality identification. The corpus contains information from 836 participants. Two modalities of each handwritten text are provided: the manually transcribed essay and the scanned image of the essay.

During the data collection phase, the researchers employed a psychological instrument, the TIPI (Ten-Item Personality Inventory) [15], to ascertain each subject's personality characteristics according to the Big Five Model. The values for each personality trait are indicated in binary form. The presence of the trait is denoted by 1 and the absence by 0. A summary of the content of the corpora related to the Big-Five model for personality recognition is shown in Table 1.

### 2.2 Automatic Personality Recognition by Text

In recent years, several research works have been carried out for automatic personality recognition from text classification, in which ML or DL techniques are used, taking some text corpus for training these methods. For this purpose, different approaches are used, such as using texts from voice transcripts or texts written by the person or published in social networks, from which it can be determined whether a person has a certain personality trait [16, 17, 18].

Majumder's work [19] shows a method where, starting from the essay's corpus (used as training data), they use a binary classification convolutional

neural network (CNN) for each personality trait with the same structure, whose purpose is to classify whether the personality trait is present or not.

The process starts with a preprocessing and filtering of the input data, where all sentences of each trial are transformed into n-gram feature vectors, followed by a feature extraction, where the vectors are concatenated with the per-word semantic features and with the François Mairesse features [20], which is a set of document-level features for personality detection.

The result is a variable length representation, which is fed into the CNN where the classification is performed. The accuracy results obtained for the personality traits are in the range of 56.71% to 62.68%.

Tandera et al. [21] conducted research based on the Big-Five personality model. This research uses the myPersonality corpus. In this work they tested traditional ML and DL methods. In ML models they used algorithms such as Support Vector Machine (SVM), Logistic Regression, Naive Bayes, Gradient Boosting and Linear Discriminant Analysis (LDA) and as DL models they tested MLP (multilayer perceptron), GRU (Gated Recurrent Unit), LSTM (long Short-Term Memory), CNN 1D (one-dimensional convolutional neural network), and a combination of the latter two (LSTM + CNN 1D).

The authors conclude that their experiments results demonstrate that deep learning techniques can enhance the accuracy of recognition models. However, they acknowledge that for certain personality traits, the obtained accuracy values were relatively low. The evaluated models achieved an accuracy in the range of 68.63% to 74.17%.

In the work of Liu et al. [22], short text personality is determined using deep learning models from the PAN 2015 Author Profiling task dataset [11], which is an English language database gathered from Twitter, containing approximately 14,000 tweets and the five personality traits labeled and evaluated each with values between -0.5 and 0.5 corresponding to 152 users.

The evaluation metric used was Root Mean Square Error (RMSE), obtaining results in the range of 0.109 to 0.167 when analyzing tweets at the user level and in the range of 0.127 to 0.189 in the analysis of individual tweets. The authors conclude that their methodology achieves results comparable to those of the state of the art in predicting personality traits.

## 3 Description of the Methodology

This section shows the methodology for the construction of the PersonText corpus. It is worth mentioning that the texts of the participants' opinions, as well as the content of the corpus is completely anonymous, respecting the privacy of the participants. To guarantee this, each text was reviewed and, in the parts where a name was mentioned, it was omitted or replaced by the following word: "$NAME$". Additionally, the architecture of the ML and DL models used to test the corpus is described.

Participants register and log on to the PersonApp platform, where they take the standardized personality test and record videos talking about a topic of their choice. The audio is then extracted from the videos and converted to text. On the other hand, the results of the standardized test are evaluated, and labels of presence and absence of each trait are assigned based on the values obtained in the test.

Finally, the texts, the generic information of the participant, the values obtained in the test and the presence and absence labels of each trait are merged to generate the corpus. Figure 1 shows the general diagram of the process used to build the corpus.

### 3.1 PersonApp

PersonApp is a multiplatform system developed internally by the research group of the Instituto Tecnológico de Culiacán that aims to collect data such as videos and IPIP standardized personality test results from participants [6].

This platform was created to evaluate the personality of each participant through a standardized personality test made up of 50 IPIP items and to obtain values for each of the five personality traits of the Big-Five model.

In addition, the platform enables participants to record videos in which they are invited to discuss a topic of their choosing. They are encouraged to

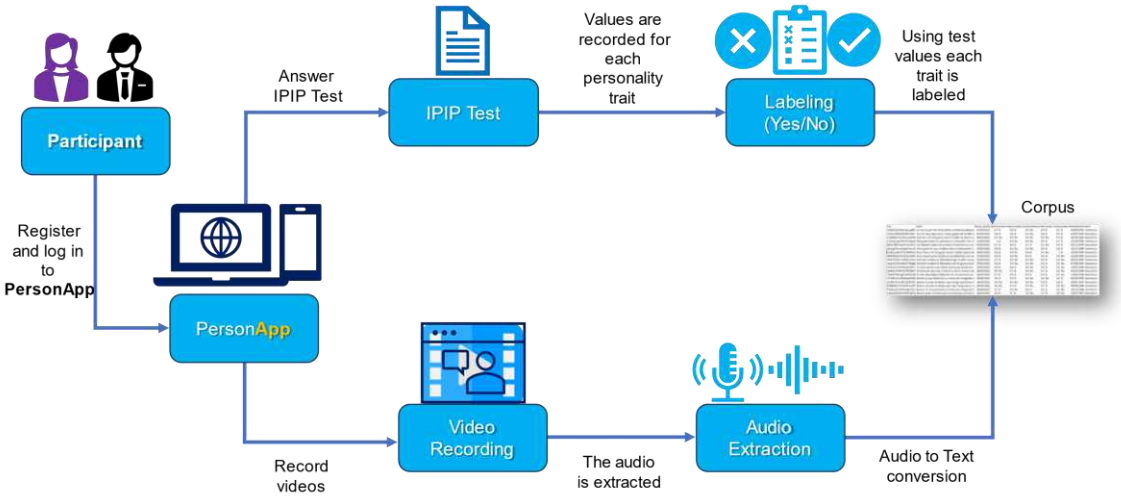



**Fig. 1.** General diagram of the construction of the PersonText corpus

imagine that they are presenting this topic to their friends in a Zoom or Teams meeting.

Another scenario is to imagine themselves in an online interview and discuss their strengths and weaknesses. Additionally, they are prompted to consider presenting a topic on a streaming platform such as YouTube, Facebook, or Twitch.

### 3.2 Compilation of Texts from the Videos

For the generation of the texts, the videos of the participants were taken from the PersonApp platform. In these videos the participants express themselves freely on various topics. About 700 videos were processed.

The extraction of the texts in the videos was carried out automatically using the Speech Recognition library [23].

Once the texts were obtained, the research team reviewed them to validate the audio-to-text conversion and correct details that occurred during the process.

In addition to the videos, the platform has all the participants' data such as name, gender, email, age, and the results of the IPIP tests consisting of the scores obtained on a scale between 0 and 1 corresponding to each of the five personality factors.

### 3.3 Structure of the PersonText Corpus

PersonText is a Spanish-language corpus containing texts of comments from different individuals, as well as their corresponding personality trait values. The inventory with the personality trait values accompanying the texts is the result of the 50-item IPIP test taken by each participant. The PersonText corpus was created in a CSV (comma-separated values) file and has 213 texts in Spanish.

The corpus data are stacked and divided into sections ranging from the user, the text of the responses of each experiment, followed by the IPIP test result values, as well as the presence label (Yes/No) of the personality trait. Table 2 shows an excerpt from the PersonText corpus.

The corpus has presence labels for each Personality trait. These were defined based on the IPIP test scores, where if the assessment result was higher than 0.6 it was labeled with a "Sí" (Yes in English), otherwise with a "No".

### 3.4 Personality Classification Models

This section describes various methods used for automatic personality recognition. It describes the preprocessing performed on the corpus, and the machine learning and deep learning models used to perform the binary (Yes/No) classification.

**Table 2.** Excerpt from the PersonText corpus

| UID | Text | TestDate | OpnV | OpnT | ConV | ConT | ExtV | ExtT | AgrV | AgrT | NeuV | NeuT | Sex | Age |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| n84XTQwKheUqUcjjd0OIusWEAtg1 | Un tema que me tiene dando vueltas la cabeza es que ya viene el día del niño y me gustaría como que hacerle un día especial al niño quisiera decorarle una pared quisiera hacer bollitos para ese día comprarle dulces que haya música para que él esté feliz quisiera que hiciéramos en esa semana actividades con el niño como salir al parque hacerle un día un pastel o algo al niño me gustaría que hiciéramos una video llamada que puedan estar | 23/04/2021 | 0.7 | Sí | 0.8 | Sí | 0.6 | No | 0.9 | Sí | 0.7 | No | Femenino | 29 |
| iXWwNMyEN5OBn4F6nMQnoqxNoki2 | Si a mí me preguntaran ¿Qué jugador de la NBA es mi favorito? pues por obvias razones es Stephen Curry es más que nada un botador hay más habilidosos en su posición como James Hardy muy buen botador creo que mucho mejor botador que Curry pero lo que hace Curry es sin esfuerzo siempre lo hace sin esfuerzo es natural mientras que Hardy es mucho más habilidoso | 04/03/2021 | 0.8 | Sí | 0.8 | Sí | 0.3 | No | 0.8 | Sí | 0.9 | No | Masculino | 26 |

### 3.4.1 Text Preprocessing

The preprocessing performed on each text includes operations such as removing Web addresses and line breaks, converting words to lowercase, omitting numbers, discarding non-alphanumeric characters (such as single quotation marks), removing blank spaces, filtering words that do not add value or meaning to the sentence (Stop Words), performing lemmatization on each word (reducing words to their root or base form), and finally removing punctuation marks.

### 3.4.2. Architecture of Machine and Deep Learning Models

The Keras library with the Python programming language was used to create the models. As a first step, the texts included in the PersonText corpus described in the previous section were preprocessed. Then, features were extracted from the corpus using the Bag of Words method. This method counts the number of times words from a given set (bag) appear in a text document, without considering grammar or word order.

Once the input data was ready, the deep learning model was configured for which different combinations of hyperparameters (such as the number of layers, the number of input and output neurons, and the type of layer in the activation function) were considered. In the latter, softmax was used since it is recommended for classification issues.

In the loss function, we worked with binary cross entropy because this function is frequently used in binary classification problems and in our research work, we seek to recognize Yes/No classes. In addition, different tests were performed with different types of optimizers, finally choosing the RMSprop optimizer, since this optimizer adapts the learning rate in small batches and with it, we obtained better results.

This research work examines several neural networks, including the multilayer perceptron (MLP). The model architecture consists of an embedding layer as the input layer, followed by a Flatten layer and then a dense layer with a ReLU activation function. Finally, as output layer it presents a dense layer with softmax activation function.

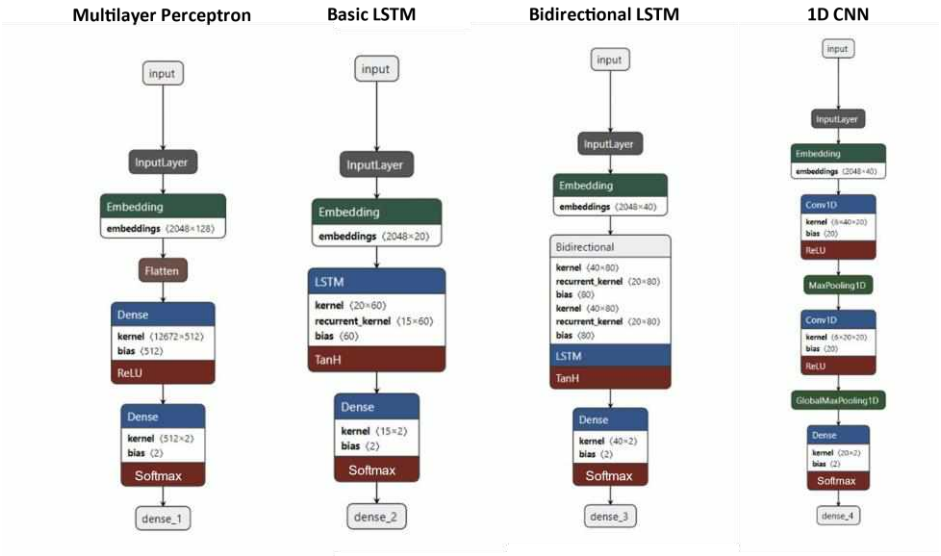



**Fig. 2.** Graphical view of the configuration of the machine and deep learning models

Training was also done with a basic LSTM neural network which is a sequential model with an input layer. The classifier receives the text in an embedding layer as input layer. In the next LSTM layer, we have 15 neurons and a dropout considered for the output of the embedding layer of 0.5. Finally, we have a dense layer with two output neurons with softmax activation function.

We also worked with a Bidirectional LSTM which is also a sequential model where the layers selected for this network are: embedding layer as input layer, followed by the Bidirectional LSTM layer, and Dense layer with softmax activation function.

Finally, a one-dimensional Convolutional Model (1D CNN) was also configured, which has an embedding layer as input, to which a 1D convolutional layer with ReLU activation is added, followed by a 1D max-pooling layer, a second 1D convolutional layer like the first one, then a global max-pooling layer and finally a Dense layer as output. This layer has two units and uses the softmax activation function to deliver probabilities for two classes.

Figure 2 shows the architectures of the 4 personality classification models used in the PersonText corpus tests.

## 4 Results and Discussion

Currently the PersonText corpus contains 213 text records with their respective labels and inventory of IPIP test results. Figure 3 shows the distribution of the classes for each personality trait. It is observed that the agreeableness trait is the most unbalanced. On the other hand, the traits of openness and responsibility are the most balanced.

For testing purposes, the PersonText corpus was divided into 75% for training and 25% for validation. As discussed in subsection 3.4.1, before performing the corpus division, preprocessing and cleaning of the corpus was previously performed. For training, 50 epochs were used, where the metric to be monitored was accuracy, keeping the best model found.

### 4.1 Results

To evaluate the PersonText corpus, different tests were performed with several machine learning and deep learning models. For this purpose, the metrics used in this research were accuracy, which represents the ratio between the number of correct predictions and the total number of input samples, and precision, which consists of determining the

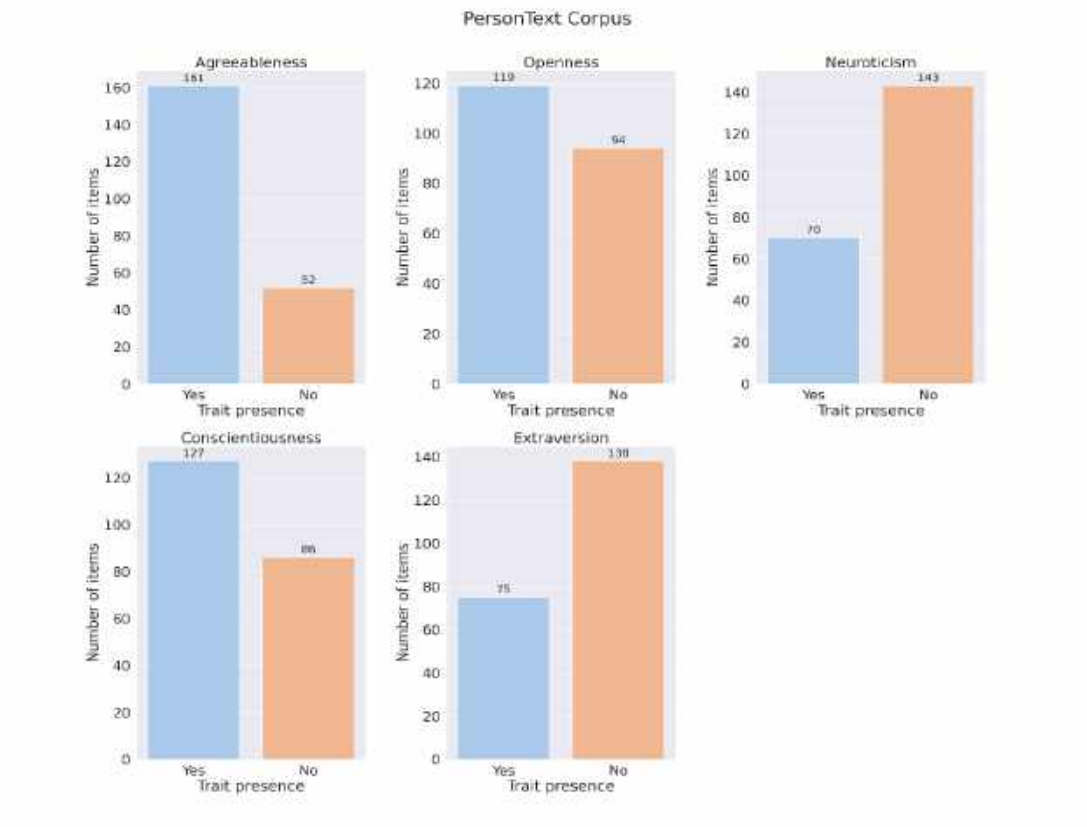



**Fig. 3.** PersonText corpus labels distribution

percentage of true positive predictions among the total resulting from adding the true positive predictions plus the false positives.

Independent models were used to determine each personality trait. For each personality trait, different configurations of machine learning and deep learning models were tested. Each type of model was configured with the same structure, i.e., the same configuration of hyperparameters in terms of number of epochs, layers, neurons, optimizer, activation, and loss functions, among others.

Table 3 shows and compares the accuracy results obtained by training different classifiers on the PersonText corpus, as well as two of the main corpora mentioned in the related work section (myPersonality and essays). Likewise, in Table 4, the values obtained with the precision metric are shown.

### 4.2 Discussion

Regarding the automatic personality recognition models assessed, it can be observed that in the case of the PersonText corpus, the most optimal results were obtained with the basic LSTM model. In both accuracy and precision, values of more than 70% were obtained, and in the five personality traits, this model obtained the best values in the metric of accuracy.

For the trait Extraversion, the Bidirectional LSTM model and the CNN model also obtained the best value, and for the trait Neuroticism, the CNN model was equal to the basic LSTM model. It can also be seen that for all traits a higher accuracy was obtained with the PersonText corpus than with the myPersonality and essays corpora.

In the precision metric, the PersonText corpus produced much higher results for the traits of

**Table 3.** Accuracy obtained from tests with different corpora

| Corpus | Classifier | Personality traits (*Accuracy*). | | | | |
|---|---|---|---|---|---|---|
| | | **Openness** | **Conscientiousness** | **Extraversion** | **Agreeableness** | **Neuroticism** |
| PersonText | MLP | 0.68519 | 0.64815 | 0.64815 | 0.68519 | 0.64815 |
| PersonText | Basic LSTM | **0.74074** | **0.74074** | **0.70370** | **0.72222** | **0.74074** |
| PersonText | Bidirectional LSTM | 0.68519 | 0.70370 | **0.70370** | 0.70370 | 0.72222 |
| PersonText | 1D CNN | **0.72222** | 0.68519 | **0.70370** | 0.70370 | **0.74074** |
| myPersonality [13] | MLP | **0.73201** | 0.58993 | 0.57914 | 0.55036 | 0.59712 |
| myPersonality [13] | Basic LSTM | 0.72302 | 0.65647 | **0.62410** | 0.60432 | **0.60432** |
| myPersonality [13] | Bidirectional LSTM | 0.72302 | **0.66727** | **0.62410** | **0.60971** | 0.60252 |
| myPersonality [13] | 1D CNN | 0.72302 | 0.55935 | 0.57914 | 0.49101 | 0.59712 |
| essays [8] | MLP | 0.54943 | 0.48298 | **0.54295** | 0.53160 | 0.52836 |
| essays [8] | Basic LSTM | **0.59968** | 0.52998 | 0.52026 | **0.55105** | **0.54781** |
| essays [8] | Bidirectional LSTM | 0.59806 | **0.53485** | 0.52674 | 0.53485 | 0.53971 |
| essays [8] | 1D CNN | 0.50405 | 0.48622 | 0.51540 | 0.50729 | 0.47164 |

agreeableness and neuroticism than for the other traits. For the trait of agreeableness, the basic LSTM model obtained a value of 0.85577, and for the trait of neuroticism, the 1D CNN model obtained a value of 0.85714.

## 5 Conclusions and Future Work

This research work presents the construction of a corpus in Spanish from texts extracted from videos obtained from the PersonApp data collection platform. These texts were related to personality scores for the traits of the Big-Five model.

To confirm their usefulness, automatic recognition models were developed using machine learning and deep learning techniques and results were obtained that allow us to answer our research question by concluding that it is possible to build machine and deep learning models that can obtain an acceptable performance when applied to a corpus in Spanish generated from textual transcripts of videos and their corresponding labels to personality traits based on a standardized personality test.

Likewise, it is concluded that the results obtained for the PersonText corpus with the models used were superior to those obtained for the other corpora reviewed.

The main contribution of our research is the methodology used for the creation of the PersonText corpus oriented to personality recognition in Spanish and the corpus itself. Although other corpora exist [5, 8, 11, 13], they are generally in languages other than Spanish. In the course of developing this research, we have discovered that the construction of a corpus is a significant challenge.

We consider the PersonText corpus to be a significant contribution to the field, as our corpus is oriented toward personality detection and the textual information it contains is derived from video recordings rather than social networks.

**Table 4.** Precision obtained from tests with different corpora

| Corpus | Classifier | Personality traits (*Precision*). | | | | |
|---|---|---|---|---|---|---|
| | | **Openness** | **Conscientiousness** | **Extraversion** | **Agreeableness** | **Neuroticism** |
| PersonText | MLP | 0.67143 | 0.32407 | 0.82075 | 0.34259 | 0.32407 |
| PersonText | Basic LSTM | **0.72931** | **0.71542** | 0.70227 | **0.85577** | 0.75556 |
| PersonText | Bidirectional LSTM | 0.69091 | 0.70821 | **0.72283** | 0.68627 | 0.69444 |
| PersonText | 1D CNN | 0.71841 | 0.65227 | 0.68393 | 0.84906 | **0.85714** |
| myPersonality [13] | MLP | **0.79378** | 0.58077 | 0.28957 | 0.55083 | 0.29856 |
| myPersonality [13] | Basic LSTM | 0.36151 | 0.65106 | **0.61460** | 0.60423 | **0.71758** |
| myPersonality [13] | Bidirectional LSTM | 0.36151 | **0.66259** | 0.61047 | **0.61249** | 0.57491 |
| myPersonality [13] | 1D CNN | 0.36151 | 0.27968 | 0.28957 | 0.24550 | 0.29856 |
| essays [8] | MLP | 0.55019 | 0.47864 | **0.54871** | 0.53372 | 0.26418 |
| essays [8] | Basic LSTM | **0.60651** | 0.52834 | 0.51999 | **0.55098** | **0.56218** |
| essays [8] | Bidirectional LSTM | 0.59806 | **0.53367** | 0.52603 | 0.53803 | 0.54223 |
| essays [8] | 1D CNN | 0.25203 | 0.24311 | 0.25770 | 0.25365 | 0.23582 |

The creation of the PersonText corpus offers an alternative for researchers wishing to tackle the task of text-based automatic personality recognition with a focus on the Spanish language. This dataset was generated from the PersonApp platform which contains information and results of standardized personality tests based on the Big-Five model.

Once the corpus was created, it was processed and subsequently tested to verify the research question. The corpus was evaluated using different neural network models to classify personality traits. Among the models, neural networks such as MLP, LSTM, Bidirectional LSTM and 1D CNN were tested, and their configuration and architecture were defined.

Five neural networks from each model were used to recognize each personality trait (Agreeableness, Openness, Conscientiousness, Extraversion and Neuroticism). The best result was 74% accuracy, which is a satisfactory result, since it is superior to those obtained by some state-of-the-art works [19] and comparable to other state-of-the-art works [21] reviewed in the related works section.

A limitation in the Spanish language is that there are not many works related to the construction of corpora oriented to automatic personality recognition based on Spanish text, since the existing data sets contain, in general, data extracted from social networks such as Facebook or Twitter in a language other than Spanish, so this research provides a corpus that can be used in future research by the Spanish-speaking scientific community.

As future work, it is proposed to increase the size of the corpus with the support of a group of expert psychologists. It is recommended to

improve the automatic recognition models by exploring strategies other than those analyzed in this research. These include the use of class balancing techniques or data augmentation, which may help to improve the accuracy and precision results of the automatic recognition models. Additionally, the optimization of hyperparameters to improve the analyzed models should be explored.

## 6 Data Availability

The corpus is available for download[1].

## Acknowledgments

The authors would like to thank the Instituto Tecnológico de Culiacán for all the support given to this research.

## References

1. **Funder, D. C. (2013).** The personality puzzle (6th edition). New York: W. W. Norton & Co. (either in hardback: ISBN: 978-0-393-91311-8, or in paperback: ISBN: 978-0-393-12441-5).
2. **Myers, I. B., Myers, P. B. (2010).** Gifts differing: Understanding personality type. Nicholas Brealey.
3. **Goldberg, L. R. (1992).** The development of markers for the Big-Five factor structure. Psychological Assessment, Vol. 4, No. 1, pp. 26–42.
4. **Stillwell, D. J., Kosinski, M. (2004).** myPersonality project: Example of successful utilization of online social networks for large-scale social research. American Psychologist, Vol. 59, No. 2, pp. 93–104.
5. **Ramos, R., Neto, G., Silva, B., Monteiro, D., Paraboni, I., Dias, R. (2018).** Building a corpus for personality-dependent natural language understanding and generation. Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC'18.
6. **Bátiz-Beltrán, V. M., Zatarain-Cabada, R., Barrón-Estrada, M. L., Cárdenas-López, H. M., Escalante, H. J. (2022).** A multiplatform application for automatic recognition of personality traits for learning environments. International Conference on Advanced Learning Technologies (ICALT), Vol. 2022, pp. 49-50, DOI: 10.1109/ICALT55010.2022.00022.
7. **IPIP (2023).** International personality item pool: A scientific collaboratory for the development of advanced measures of personality traits and other individual differences. Recover from http://ipip.ori.org/.
8. **Pennebaker, J. W., King, L. A. (1999).** Linguistic styles: language use as an individual difference. Journal of personality and social psychology, Vol. 77, No. 6, 1296–1312. DOI: 10.1037/0022-3514.77.6.1296.
9. **John, O. P., Donahue, E. M., Kentle, R. L. (1991).** The big-five inventory-version 4a and 54. Berkeley, CA: Berkeley Institute of Personality and Social Research, University of California.
10. **Kim, L., Walter, D. (2008)**. Personae: A corpus for author and personality prediction from text. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco, European Language Resources Association (ELRA).
11. **Rangel-Pardo, F. M., Celli, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W. (2015).** Overview of the 3rd author profiling task at PAN 2015. CLEF 2015 Evaluation Labs and Workshop Working Notes Papers, pp. 1– 8.
12. **Rammstedt, B., John, O. P. (2007).** Measuring personality in one minute or less: A 10-item short version of the big five inventory in english and german. Journal of Research in Personality, Vol. 41, No. 1, pp. 203–212. DOI: 10.1016/j.jrp.2006.02.001.
13. **Celli, F., Pianesi, F., Stillwell, D., Kosinski, M. (2013).** Workshop on computational

[1] https://catalabs.mx/datasets/persontext/

personality recognition: Shared task. AAAI Workshop - Technical Report.

14. **Ramírez-de-la-Rosa, A. G., Villatoro-Tello, E., Jiménez-Salazar, H. (2018).** HWxPI: A multimodal spanish corpus for personality identification. Latin American and Iberian Languages Open Corpora Forum.
15. **Gosling, S. D., Rentfrow, P. J., Swann, W. B. (2003).** A very brief measure of the big-five personality domains. Journal of Research in Personality, Vol. 37, No. 6, pp. 504–528. DOI: 10.1016/S0092-6566(03)00046-1.
16. **Xue, D., Wu, L., Hong, Z., Guo, S., Gao, L., Wu, Z., Zhong, X., Sun, J. (2018).** Deep learning-based personality recognition from text posts of online social networks. Appl Intell Vol. 48, pp. 4232–4246. DOI: 10.1007/s10489-018-1212-4.
17. **Rissola, E. A., Bahrainian, S. A., Crestani, F. (2019).** Personality recognition in conversations using capsule neural networks. IEEE/WIC/ACM International Conference on Web Intelligence pp. 180–187.
18. **Tadesse, M., Lin, H., Xu, B., Yang, L. (2018).** Personality predictions based on user behavior on the facebook social media platform. IEEE Access, pp. 1–1. DOI: 10.1109/ACCESS.2018.2876502.
19. **Majumder, N., Poria, S., Gelbukh, A., Cambria, E. (2017).** Deep learning-based document modeling for personality detection from text. IEEE Intelligent Systems, Vol. 32, pp. 74–79. DOI: 10.1109/MIS.2017.23.
20. **Mairesse, F., Walker, M., Mehl, M., Moore, Roger. (2007).** Using linguistic cues for the automatic recognition of personality in conversation and text. Journal Artificial Intelligence. Research, Vol. 30. pp. 457–500. DOI: 10.1613/jair.2349.
21. **Tandera, T., Hendro, Suhartono, D., Wongso, R., Prasetio, Y. (2017).** Personality prediction system from facebook users. Procedia Computer Science, Vol. 116, pp. 604–611. DOI: 10.1016/j.procs.2017.10.016.
22. **Liu, F., Perez, J., Nowson, S. (2016).** A recurrent and compositional model for personality trait recognition from short texts. Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES), The COLING 2016 Organizing Committee, Osaka, Japan, pp. 20–29.
23. **Zhang, A. (2015).** Speech recognition (Version 2.1) [Software]. Available from https://github.com/Uberi/speech_recognition#readme.

*Article received on 26/05/2023; accepted on 07/06/2024.*
**Corresponding author is María Lucía Barrón-Estrada.*

ISSN 2007-9737

# Reinforcement Learning Based Fog and Cloud Resource Allocation for an IoRT-aware Business Process

Najla Fattouch[1,*], Imen Ben-Lahmar[2], Khouloud Boukadi[1]

[1] University of Sfax,
Faculté des Sciences Économiques et de Gestion,
Tunisia

[2] University of Sfax,
Institut Supérieur d'Informatique et de Multimédia,
Tunisia

fattouchnajla@gmail.com, imen.benlahmar@isims.usf.tn, khouloud.boukadi@fsegs.usf.tn

**Abstract.** IoRT-aware BP aims to promote the business process (BP) within robotics and IoT capacities. This incorporation ensures machine-to-machine (M2M) communication and the automatic execution of tasks by using robot devices. Nonetheless, the execution of this process inside the enterprise may be costly due to the consumed resources, the need for computational capacity, etc. To close these gaps, the business process outsourcing (BPO) strategy can be carried out to outsource the IoRT-aware BP to external environments (e.g., Cloud, Fog, etc.). To profit from outsourcing, an enterprise should identify suitable resources to ensure optimal process execution. The selection of resources is known in the literature as resource allocation (RA). The RA problem is described in this work using the Markov Decision Process (MDP), and it is resolved using reinforcement learning (RL). The proposed approach relies, on one hand, on Q-learning as an RL algorithm, and on the other hand, it considers the extension of the ifogSim tool to support the process execution using Fog and Cloud resources. The obtained results are promising in terms of response time regarding the scale-up of the considered resources. Furthermore, the experimental results show that our approach offers a substantial advantage in optimizing the performance of RA, which confirms its usefulness and relevance compared to other common methods.

**Keywords.** Resource allocation, reinforcement learning, IoRT-aware business process, fog, cloud.

## 1 Introduction

Industry 4.0 is becoming more and more popular as smart devices and Internet of Things (IoT) technologies proliferate. By merging the current manufacturing system with the industry technology system, this new paradigm aims to improve the production system [17]. Industry 4.0 gives birth to various technologies, such as the Internet of Robotic Things (IoRT). It combines IoT and robotics into one technology, which is called the IoRT. ABI Research [29] came up with the idea of the IoRT, which defines it as a collection of intelligent, disparate devices that can manage events and massive amounts of data.

Numerous researchers aim to leverage IoRT paradigm to incorporate it into other domains (e.g., Business Process (BP), etc.). Integrating IoRT technology into the traditional BP creates the IoRT-aware BP generation [4], which strives to increase productivity, automate operations, etc. However, executing an IoRT-aware BP within the organization could be expensive due to the energy and resources consumed, among others. To bridge these gaps, the Business Process Outsourcing strategy (BPO) remains the best solution by executing some parts of the entire process outside the enterprise [10].

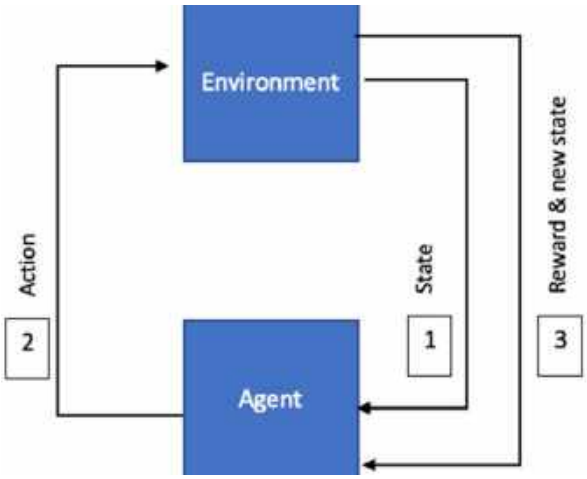



**Fig. 1.** Reinforcement learning process [14]

Numerous environments can be considered to perform process outsourcing, for instance, the Cloud and Fog. The Cloud is defined as hardware resources and software services available on the internet [32]. It has a large storage capacity compared to other surroundings [1]. Furthermore, it enables businesses to increase their services in response to client demand gradually.

Using the Cloud during the execution of IoRT-aware BPs allows the business managers to meet their process's computational and availability requirements. Despite its benefits, the Cloud fails to support sensitive applications due to the distance between the user devices and the Cloud data center. Therefore, the Fog environment appeared.

It is a paradigm that brings computational resources and services to the network edge near user devices, lowering latency and connecting with Cloud resources [2]. It is characterized by its ability to perform latency-sensitive applications regarding its proximity to the user's devices (e.g., IoT, robots, etc.).

To achieve the outsourcing of an IoRT-aware BP into Cloud and Fog environments, a business manager should allocate the appropriate resources of ($Cloud_i$, $Fog_j$ where i,j $\in [1..n]$). The Resource Allocation (RA) is defined as the groundwork for BP outsourcing.

It must match the demand of the instances of processes running with the resources available for a specific business objective (e.g., minimizing process cost, maximizing process availability, etc.) [28]. Thus, the RA will allow the business managers to ensure the effectiveness of their processes. Additionally, it enables businesses to increase production, realize an equitable distribution of responsibilities, and decrease the need for human intervention.

Back to the literature, several approaches addressed RA for the business process. In general, these approaches focus on the Cloud resources [11], or Fog ones [19, 31]. However, considering both environments enables business managers to take advantage of both. Furthermore, most of these approaches deal with process cost [23, 25, 22] and execution time [15, 12].

Nonetheless, reducing the energy consumption within the enterprise and reducing latency can improve the selection of resources. Furthermore, diverse algorithms were applied for supporting RA in the BPs field, including exact approaches [11], meta-heuristic methods (e.g., [12]), and machine learning [3]. However, these algorithms demonstrate serious overhead and the need for more accuracy in their evaluation.

Therefore, to close the gaps in the literature-based RA approach, we take advantage of Reinforcement Learning (RL) as an Artificial Intelligence (AI) algorithm to perform resource allocation with less overhead and scattered errors [19]. We aim through this paper to propose a Reinforcement Learning-based approach for the Fog and/or Cloud RA to achieve an optimal execution of the IoRT-aware BP. Our proposal aims to satisfy a set of different RA goals. Therefore, it intends to minimize the RA cost and reduce the energy consumed.

In addition, our proposal intends to reduce the execution time and the latency value. To accomplish these objectives, we extended the ifogSim tool to estimate the execution cost, consumed energy, execution time, and latency of an IoRT-aware BP using Fog and Cloud resources. The obtained results are promising. Furthermore, compared to other RA methodologies, the experimental results prove the

**Table 1.** Comparison of a set of RA-based approaches regarding the identified criteria (part 1)

| Year | Paper | BP Type | Resource Type | | Used Algorithm | | | |
|---|---|---|---|---|---|---|---|---|
| | | | Fog | Cloud | Exact | Heuristic | meta-heuristic | RL |
| 2021 | | Classic BP | - | - | - | - | - | - |
| 2022 | | Classic BP | - | - | - | x | - | - |
| 2022 | | Classic BP | - | - | - | - | x | - |
| 2021 | | Classic BP | - | x | | - | - | - |
| 2022 | | Classic BP | - | - | - | - | - | x |
| 2021 | | IoT-aware BP | x | x | - | - | - | - |
| 2020 | | Classic BP | - | - | - | - | - | x |
| 2020 | | Classic BP | - | - | - | - | - | x |
| 2020 | | - | - | x | - | - | - | - |
| 2022 | | - | x | - | - | - | - | - |
| 2024 | | Classic BP | - | - | x | - | - | - |
| 2024 | | - | - | - | - | - | - | - |
| 2024 | | Classic BP | - | x | - | x | - | - |
| 2022 | | - | - | x | - | - | - | - |

**Table 2.** Comparison of a set of RA-based approaches regarding the identified criteria (part 2)

| Paper | Used criteria | | | | Simulation tool | RL support | Workflow patterns | Granularity |
|---|---|---|---|---|---|---|---|---|
| | Cost | Energy | Execution time | Latency | | | | |
| | - | - | - | - | Simulation engine | semi-automatic | - | task |
| | - | - | x | - | TypeScript and MongoDB | automatic | x | task |
| | - | - | x | - | GRINGO 5.5.0, I-DLV 1.1.6 | automatic | x | task |
| | x | - | - | - | CloudSim | automatic | x | task |
| | x | - | - | - | Apromore | semi-automatic | - | task |
| | x | - | x | - | - | automatic | x | task |
| | x | - | - | - | - | automatic | - | task |
| | x | - | - | - | - | automatic | - | task |
| | - | x | - | - | - | automatic | - | task |
| | - | - | - | x | ifogSim | automatic | - | task |
| | x | - | x | - | Prosimos | automatic | x | task |
| | - | - | - | - | - | automatic | - | task |
| | x | x | - | - | - | automatic | - | task |
| | - | x | x | - | Cloudsim | automated | - | task |

efficacy of the proposed RL-based RA approach by offering a significant benefit in optimizing RA performance. The remainder of this paper is structured as follows. Section 2 presents a background that briefly describes the IoRT-aware BP and RL algorithm. Section 3 overviews the recently published approaches that deal with the RA question. Section 4 details the proposed reinforcement learning-based approach for RA to ensure the adequate execution of an IoRT-aware

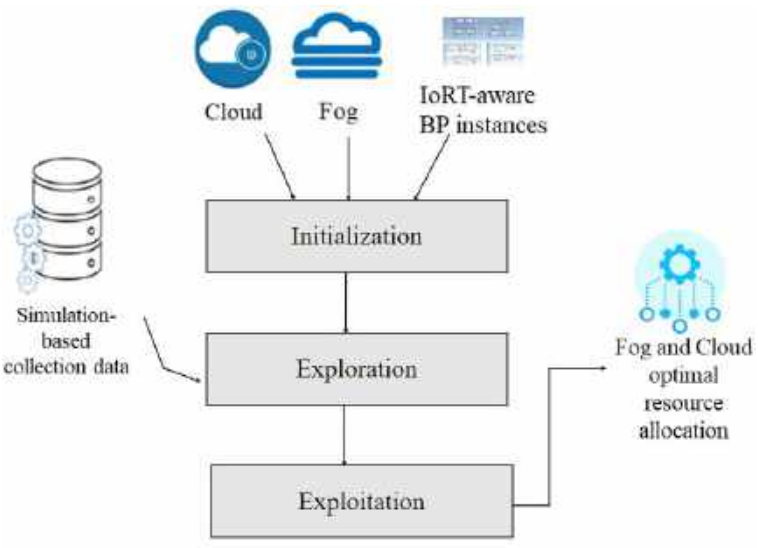



**Fig. 2.** Proposed approach for fog and cloud resources allocation

BP. The evaluation and discussion of the proposed approach are presented in Section 5. Finally, Section 6 summarizes our work and highlights its future directions.

## 2 Background

This section presents some relevant concepts to ensure the execution of an IoRT-aware BP using suppliers' resources. In this setting, we intend to give an overview of an IoRT-aware BP. Then, we present a brief description of the RL algorithm.

### 2.1 IoRT-aware Business Process

In recent years, the fourth industrial revolution has led to the development of several new technologies, such as the Internet of Robotic Things (IoRT). The IoRT is defined as the improvement of the Internet of Things (IoT) itself, where robotic technology has been embedded in IoT, Cloud, and Networking [18]. The IoRT is characterized by heterogeneous advantages that make it among the most attractive technologies.

In [29], the authors classified the IoRT abilities into four main categories: basic, high-level, interaction, and system-level abilities.

Among the basic IoRT abilities, we note that it has a broader horizon in time, space, and information type regarding the integrated sensors within the devices.

Moreover, the robots' ability to move independently is considered one of the basic IoRT advantages. In the IoRT higher-level abilities setting, we denote the capacity of the robot devices to automatically make the right decisions and distinguish the best course to meet its missions.

Furthermore, robot devices use AI techniques to improve their decisions. In the robot interaction ability context, the robots are characterized by their ability to interact with users and other systems. Besides, regarding their experience and reasoning information, the robots ensure communication between themselves, things, and their environment.

Nonetheless, in the system-level abilities setting, we cite its capability to be customized and configured for particular tasks. Regarding the advantages mentioned above, the IoRT technology sweeps several fields, for instance, agriculture, health, etc.

In this setting, business managers seek to benefit from this technology to automate their processes. Therefore, several researchers seek

**Algorithm 1** Reinforcement learning-based algorithm for optimal Fog and Cloud resource allocation

**Require:**

$N$ - Number of IoRT-aware BP
$A_1 = \{Fog_1, \ldots, Fog_n\}$ - Action set for Fog
$A_2 = \{Cloud_1, \ldots, Cloud_m\}$ - Action set for Cloud
$TR = 0$, $R = 0$, $R' = 0$
$\alpha = 0.5$, $\gamma = 0.9$

**Ensure:** Optimal resource allocation for each process task

**while** $i \in [1...N]$ **do**
    Give coarse-grained decision ($D$)
    Give the SESE tasks $(T_1, \ldots, T_n)$
    Give the SESE workflow patterns $(WP)$
    Fixed $S$ state using SESE tasks
    **if** $(D ==$ Fog$)$ **then**
        Resource = $A_1$
    **else if** $(D ==$ Cloud$)$ **then**
        Resource = $A_2$
    **else if** $(D ==$ Fog & Cloud$)$ **then**
        Resource = $A_1 + A_2$
    **else if** $(D ==$ in-house$)$ **then**
        No allocation done
    **end if**
    **if** $(D \neq$ in-house$)$ **then**
        Initialize Q-table[S, Resource]
        Resource_allocation_function(S, Resource, WP)
    **end if**
**end while**

to integrate the IoRT technology within the BP; we cite among them [27], where the authors define the integration of the IoRT within the BP as the automation of the BP to resolve a complicated situation.

Moreover, in [21], authors the integration as a form of enterprise digitization through the automation of its process tasks. According to [4], integrating the IoRT within the BP gives the newest process generation called IoRT-aware Business Process.

The latter aims to automate the classic BP, where robots and IoT devices are used to perform process tasks rather than humans. Therefore, the automation of process tasks allows firms to eliminate the burden of human errors, speed up their production, improve their productivity, etc. Furthermore, embedding the IoRT technology within the classic process allows enterprises to reduce costs and eliminate the burden of recruiting employers.

### 2.2 Reinforcement Learning

Reinforcement Learning (RL) is an AI model that allows algorithms to learn from their trials and errors. Intending to understand how the algorithm makes the right decision, the RL has been confronted with a set of decisions [30].

Therefore, whether it has taken the wrong decision, it is penalized, while, in the case that the right decision is taken, the algorithm gains a reward. The problems in the RL are frequently formulated as a Markov Decision Process (MDP). The MDP is defined as a mathematical framework for solving decision-making problems [33].

It is used to present the environment model and dynamicity [14] (see Figure 1). The MDP model usually has a transition function to transform the environment from one state to another after applying an action using an agent. According to [33], the RL has five main components which are detailed in what follows:

– $S$: Gives the set environment states.
– $A$: Presents the possible actions taken by the agent to change the environment state.
– $R(A)$ : S * A : Indicates the reward's current scale.
– $f$: Designates the state transition function. f(s,a,s') = P(s'| s,a) is the probability that state $s$ transits to $s'$ after performing an action $a$.
– $\gamma$: Is a value in the range [0, 1]. It is among the MDP parameters. It defines a discount factor that is intended to reduce the impact of future rewards on the present.

There are two main types of reinforcement models:

**Algorithm 2** Resource_allocation_function (S, Resource, WP)

**Require:** initialize $Q[S, Resource]$ with considering the (WP)

**if** T is the first task **then**
  Agent select action(a) from (Resource)
  Agent observes reward (R)
  $Q[Si, A_1] = Q(Si, Resource) + \alpha[R + \gamma * max_{Resource'}(Q(S', Resource\hat{a}) - Q(Si, Resource)]$
  TR = TR + R
**end if**
**while** $T \in [T_2...T_n]$ **do**
  Agent selects Si according to our policy (p)
  Agent select (a') from (Resource)
  Agent observes reward (R')
  $Q[Si, A_1] = Q(Si, Resource) + \alpha[R + \gamma * max_{Resource'}(Q(S', Resource\hat{a}) - Q(Si, Resource)]$
  TR = TR + R'
**end while**

– Model-free: Means that the model is based on optimal policies and value functions of the obtained data regarding the agent's interaction with the environment. For the model-free, the agent learns with trial and error from experiencing explicit [33].

– Model-based: Refers to learning optimal behavior from a model of the environment, taking actions, and observing the outcomes that include the next state and the immediate reward. The policy of a model-based can be discovered using various planning techniques [33].

In our work, we deal with the model-free as it does not require a model from the environment to make its decisions compared to the model-based.

In fact, the model-free is characterized by its simple implementation compared to the model-based that requires a lot of memory and computation.

Moreover, the model-free learns more quickly as it does not require a large amount of data to learn an optimal policy.

## 3 Related Work

In this section, we provide an overview of some recently published approaches that deal with the RA for a BP. These approaches are examined while relying on a set of relevant criteria, which are listed below:

– Business Type: This criterion lets us pinpoint the process type that is used to ensure its execution through the allocation of internal and external resources. Two types of process are addressed: a classic BP where its tasks are achieved by the human. However, an automated process gives the process that embeds one or more technologies.

– Granularity: It identifies the processing granularity used in RA. It may be a process or a sub-process that contains one or more tasks.

– Used resources: Numerous resources can be addressed to ensure the process's execution. During this work, we are interested in the Cloud and Fog resources. In fact, the Cloud resources are considered regarding their storage and processing capacities. Nonetheless, Fog ones are selected regarding their closer to the end devices for instance, sensors, cameras, and so on.

– Used algorithm: It identifies the algorithms used to accomplish the RA goal. Referring to the literature, the RA problem can be solved using mainly exact methods, heuristic algorithms, and meta-heuristic algorithms. Moreover, numerous approaches address the RA using artificial intelligence (AI) algorithms.

– Used parameters: This criterion consists of a set of heterogeneous parameters that are considered to perform the RA decision. Our extensive literature review identified the consumed energy, cost, latency, and execution time as critical RA criteria. The energy criterion depicts the amount of consumed energy to perform such task. The cost specifies the fee that must be paid to accomplish a task using such a resource. The latency is among the criteria corresponding to the needed time to transfer data from the task to the used resource.

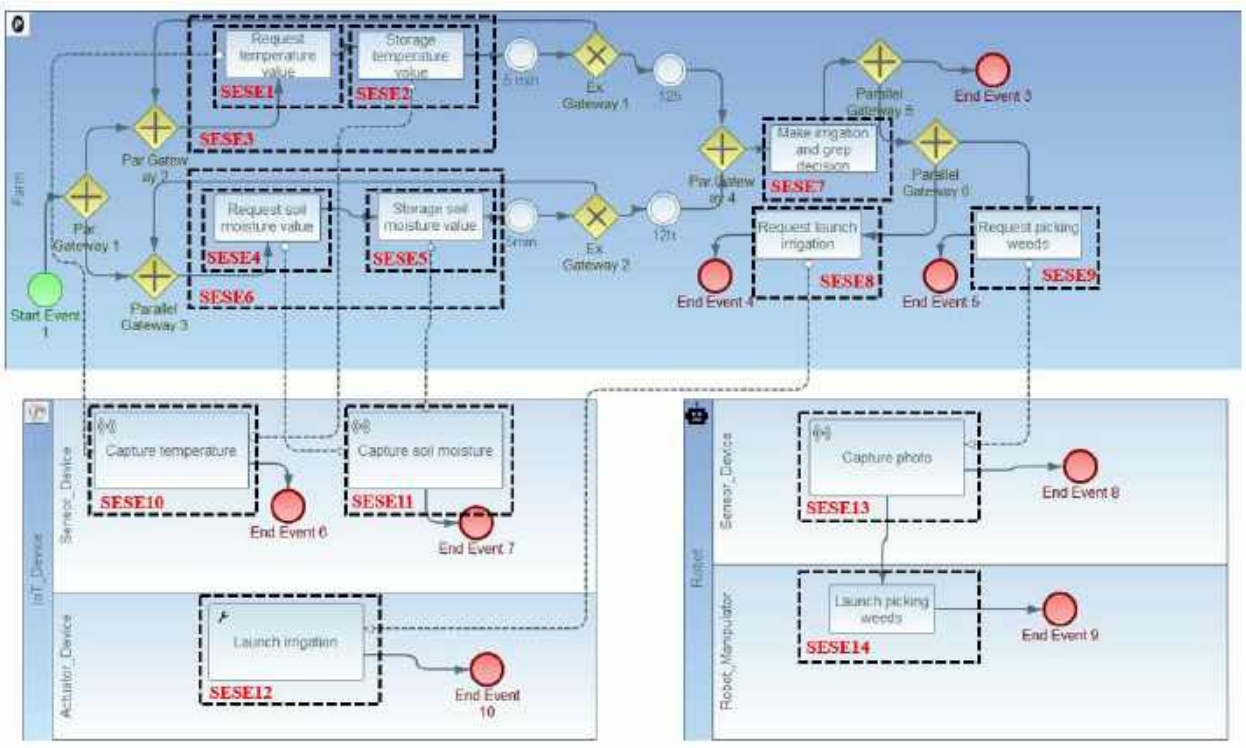



**Fig. 3.** IoRT-aware business process on agriculture field composed by fourteen SESE fragments

However, the execution time sets forward the required time to execute a task using a resource.

– RA support: Identifies whether the studied approaches deal with automatic or semi-automatic RA. It distinguishes approaches that refer to systems or processes to allocate resources without human intervention.

– Workflow patterns: Shows the dependency execution between the process tasks. Back to the literature, numerous patterns exist in the BP area, for instance, sequential flow, parallel split, etc. This criterion identifies which approaches consider workflow patterns between tasks during the RA.

– Simulation tool: Simulation is among the techniques that allow business designers to represent reality and generate hypothetical process instances. Different simulation tools have been developed in the literature to support the BP execution (e.g., ifogSim, CloudSim, etc.). This criterion distinguishes the most considered simulation tool.

Tables 1 and 2 classify the studied approaches regarding the selected criteria. Referring to table 1, we remark that most of the recent approaches, for instance, [35], [15], [12], [11], [23], [7], and [20] address the RA for the classic BP. To our knowledge, no work has been proposed for allocating resources to accomplish IoRT-aware BP execution.

Moreover, from the same table, we note that some of the studied approaches, for instance, [11] deal with the Cloud resources for process execution. However, we noticed a lack of approaches that used both Cloud and Fog resources to execute the process. Furthermore, according to table 1, we denote that few works cope with the AI algorithms to support the RA. Indeed, AI has become fundamentally ingrained within numerous fields, for instance, the BP, and so on.

**Table 3.** The SESE requirements on MIPS, RAM, size, and BW

| SESE Fragments | MIPS | RAM | Size | BW |
|---|---|---|---|---|
| SESE1 | 10 | 8 | 60 | 10 |
| SESE2 | 20 | 32 | 80 | 20 |
| SESE3 | 30 | 40 | 140 | 20 |
| SESE4 | 15 | 10 | 75 | 20 |
| SESE5 | 30 | 18 | 100 | 40 |
| SESE6 | 45 | 28 | 175 | 40 |
| SESE7 | 100 | 25 | 400 | 50 |
| SESE8 | 80 | 20 | 500 | 10 |
| SESE9 | 120 | 38 | 300 | 23 |
| SESE10 | 45 | 5 | 500 | 60 |
| SESE11 | 145 | 30 | 120 | 70 |
| SESE12 | 120 | 2 | 150 | 65 |
| SESE13 | 155 | 32 | 250 | 20 |
| SESE14 | 60 | 8 | 100 | 23 |

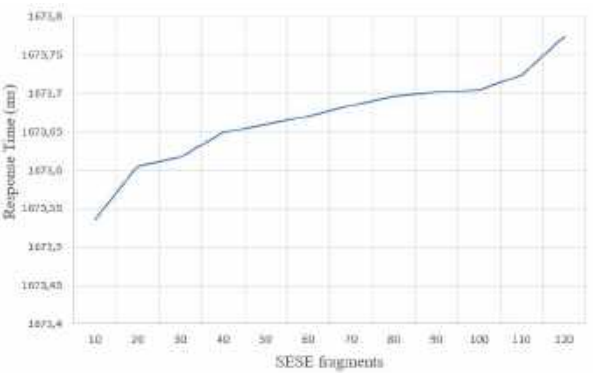



**Fig. 4.** Response time of the RL-FCRA according to the number of the considered SESE fragments

Referring to table 2, we note that most of the existing approaches (e.g., [12], [25], and [22]) deal with cost as a primary criterion to select the appropriate resources. Nevertheless, several other criteria can impact the selection of adequate resources. Therefore, the resource's energy consumption can be tackled for the RA decision, where the business manager always seeks to execute their process using resources that consume less energy. Additionally, the execution time is among the relevant criteria that can impact the RA decision, where the business managers attempt to gain time by using resources that require less time to execute their processes. Furthermore, the latency can be considered when identifying suitable resources.

Our literature study points out that several approaches (e.g., [23, 16, 25, 22]) ignore the process workflow during the RA decision-making. However, to ensure the correct process execution, it seems crucial to capture dependency between its tasks. In the same setting, we outlined from our literature review that different methods address the RA decision using process task, such as [35, 15, 12, 11, 23, 16, 25, 22] rather than the sub-process.

However, addressing resource allocation for a sub-process ensures the preservation of task workflow integrity. To bridge the distinguished RA gaps, we introduce an innovative AI-based RA methodology to proficiently execute an IoRT-aware BP leveraging Cloud and Fog resources. Our proposed approach tackles the IoRT-aware BP by dividing it into Single Entry Single Exit (SESE) fragments, which are blocks comprising one or more tasks.

Moreover, it manages process workflow to uphold task dependency execution integrity. This methodology is rooted in Reinforcement Learning (RL), capitalizing on its advantages to optimize performance and adaptability. The proposed approach defines several goals including reducing the RA cost, consumed energy, execution time, and latency.

## 4 Reinforcement Learning Based Resource Allocation Approach

Achieving an optimal RA using hand-coded heuristics and fixed strategies is difficult due to the heterogeneity of processes and resource requirements. In this research work, we design and implement a decision-making approach based on an artificial intelligence algorithm to accomplish the selection of adequate Fog and Cloud resources.

The proposed approach is called Reinforcement Learning-based Fog and Cloud

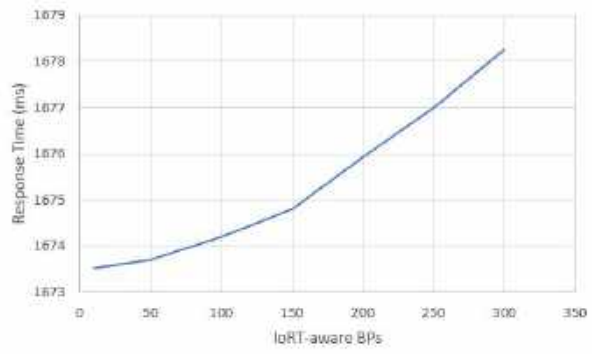



**Fig. 5.** Response time of the RL-FCRA according to episode number

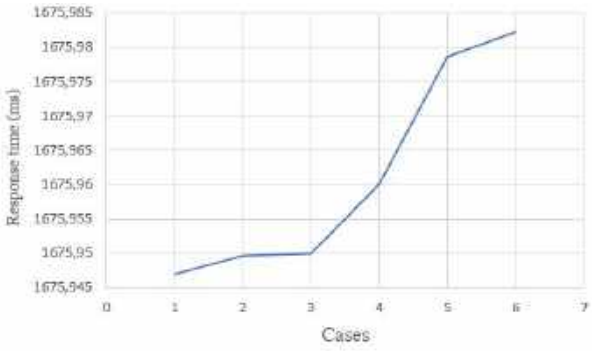



**Fig. 6.** Response time of the RL-FCRA considering various fog and cloud resources

Resource Allocation, which is referred to as (RL-FCRA). Figure 2 illustrates an overview of the suggested RL-FCRA.

It is based on the RL algorithm, and it has several advantages that have proven worthwhile in the RA field. It does not require a large data set compared to the other AI algorithms. Furthermore, RL does not require training as it can automatically adapt to new environments.

In this work, we adopted Q-learning as an RL model-free algorithm. It handles issues with stochastic transitions and reward values. In the following, we describe the proposed approach steps.

### 4.1 Initialization

Initialization refers to setting the initial values of the model parameters before it is trained. In the RL setting, the initialization phase can have a relevant impact on the algorithm's performance.

The initial values of the algorithm parameters can determine how the algorithm achieves its goals and who can handle the newest situations. Our proposal has as input an IoRT-aware BP divided into a set of SESE fragments. Each fragment includes one or more tasks.

The SESE tasks have to be executed using the Fog and/ or Cloud resources, where each resource has its initialized parameter values of cost, energy, execution time, and latency. To initialize the criteria mentioned above, for each resource, we propose to simulate the IoRT-aware BPs execution using a simulation tool.

Simulation is among the techniques that allow business designers to represent reality in a clarified manner and generate hypothetical process instances. In the literature, various simulator tools with different objectives are developed.

According to our literature exercise, we revealed that ifogSim [9] is among the most used tools for the simulation of Fog and Cloud environments. It is an open-source, java-based tool that allows easy modeling of Fog and Cloud [9].

Moreover, it enables the simulation of the process execution using Fog/ Cloud resources under different scenarios and conditions [9]. In the following, we briefly describe the considered criteria and how to estimate their values:

– Cost: Reducing expenses stands out as one of the compelling incentives driving enterprises to leverage external providers' resources, including services, platforms, and infrastructures, for executing their processes. According to the authors in [8], the use of external resources for process execution is primarily guided by overhead expenses, with careful consideration given to both the processes and resources to gauge potential cost savings.

  Therefore, we consider the equation 1 to estimate the execution cost using Fog or Cloud resource, where the $CC$ presents the current

**Table 4.** Scalability of the fog and cloud resources

| Case | Fog resources | Cloud resources |
|---|---|---|
| Case1 | 3 | 2 |
| Case2 | 5 | 5 |
| Case3 | 10 | 10 |
| Case4 | 15 | 15 |
| Case5 | 20 | 20 |
| Case6 | 30 | 30 |

cost, the $R\_PM$ gives the resource rate cost per MIPS, and the $T\_Mips$ presents the total MIPS of a SESE task:

$$\text{Cost} = CC + (R\_PM * T\_\text{Mips}). \quad (1)$$

– Energy: Depicts the energy consumed to perform the execution of process fragments using Fog and/ or Cloud resources. The execution of an IoRT-aware BP within the enterprise can be costly in consuming energy regarding the used devices and equipped resources. Consequently, business managers aim to achieve the execution of their processes with less energy consumption using external resources. To estimate the consumed energy of the Fog or Cloud resource to perform a process task, we propose equation 2, where the $CE$ represents the current energy consumed by the resource and $T$ gives the current time:

$$\text{Energy} = CE + (T * \text{Power}). \quad (2)$$

– Execution_Time: Saving time is among the significant factors encouraging enterprises to execute their processes using external suppliers' resources. These latter ones allow business managers to speed up the execution of their process where these resources replace human intervention. To estimate the execution time of a process fragment on a Fog or a Cloud resource, we propose the equation 3, where $ES$ gives the end simulation time $SS$ presents the start simulation time:

$$\text{Execution\_time} = ES - SS. \quad (3)$$

– Latency: The required time to transfer data from the process task to such resource aligns with the concept of latency. This makes it a crucial criterion for performing the IoRT-aware BP execution.

Given that IoRT involves tasks encompassing both IoT and robotics, which can be particularly sensitive to latency, it is imperative to include latency as a key consideration.

To gauge the latency value for a process task on a Fog or Cloud resource, we consider equation 4, where $\alpha$ presents the tuple CPU execution delay for sending a request to a resource and $\sigma$ gives the time to execute the task on the resource. However, $\varphi$ gives the time taken to display the information to the end-user device after processing at the Fog or Cloud node:

$$\text{Latency} = \alpha + \sigma + \varphi. \quad (4)$$

### 4.2 Exploration

In the RL, exploration involves trying different actions to discover which ones lead to the best reward. It is considered one of the relevant RL phases, allowing the agent to achieve their goals with the best reward.

We model the RA problem using the MDP which is considered the most popular mathematical framework for solving RL problems. The MDP defines the tuple {S, A, P, R}, where:

– S: Presents the environment states. In our work, we deal with tasks of the SESE fragments as a set of environment states (see equation 5):

$$S = \{T1, ..., Tn\}. \quad (5)$$

– A: Depicts the actions that can be applied to the environment. In our work, we deal with selecting the resource as an action. Consequently, we can perform the selection of a Fog resource (see equation 6) and/ or Cloud resource (see equation 7):

$$A1 = \{Fog_1, ..., Fog_f\}, \quad (6)$$

$$A2 = \{\text{Cloud}_1, ..., \text{Cloud}_c\}. \quad (7)$$

**Table 5.** Fog and cloud resources features

| Resource | MIPS | RAM | Up BW | Down BW | Rate Per MIPS | Busy Power | Idle Power |
|---|---|---|---|---|---|---|---|
| Fog1 | 155 | 50 | 200 | 100 | 0,0000006 | 0,1 | 0,001 |
| Fog2 | 200 | 40 | 300 | 110 | 0,000001 | 0.2 | 0.1 |
| Fog3 | 300 | 80 | 600 | 200 | 0.0000002 | 0.7 | 0.01 |
| Cloud1 | 100 | 40 | 400 | 100 | 0,00000003 | 100 | 70 |
| Cloud2 | 1000 | 500 | 10000 | 1000 | 0,000000005 | 200 | 200 |
| Cloud3 | 1500 | 750 | 900 | 900 | 0,0000004 | 750 | 310 |

– P: Sets forward the policy that is considered for an RL algorithm. To achieve the Fog and Cloud RA goal, we propose our policy that allows the agent to receive an SESE coarse-grained decision $D$ generated by the Multi-Criteria Decision Method (MCDM) approach presented in [6]. The coarse-grained decision gives the selected environment (e.g., Fog, Cloud, Fog&Cloud, in-house).

If the decision $D$ equals *Fog*, the agent will choose actions from the set of Fog resources $A1$. In the case where the selected decision $D$ is equal to *Cloud*, the agent will target the Cloud resources $A2$ to choose the adequate resources for the SESE tasks. However, where the decision $D$ equals *Fog&Cloud*, the agent will select the resources from both resource sets. Finally, where the *in-house* gives as the coarse-grained, the agent refrains from selecting any resources.

After identifying the set of resources, the agent initializes its Q-table values and applies algorithm 2 to both update the Q-table and select the appropriate resource. Based on algorithm 2, the agent selects the first task for each SESE and distinguishes the adequate resource according to the estimated reward $RT$ value. After that, it performs the other SESE tasks, which aim to optimize the reward value as detailed in equation 8.

Hence, it is relevant to note that to achieve the RA goal, the agent takes into account the workflow patterns between tasks. As required by the process's functional requirements, workflow patterns give the execution dependencies of the process tasks. During this work, We rely on the de facto BPMN standard [24] to consider the process workflow patterns (e.g., sequence, parallel, loop, etc.). After each iteration, the agent updates the Q-table values. The job repeats until the number of episodes (processes) is reached (see algorithm 1).

– R: Gives the reward function of the proposed RL-FCRA. Our proposed reward function (see equation 8) relies on the cost $C$, energy $E$, execution time $ET$, and latency $L$. When selecting the resource, we aim to reduce its costs and consumed energy. Moreover, we intend to minimize the latency and execution time to speed up the resource selection:

$$R = 2 * C + 2 * E + (ET + L) * (1/2). \quad (8)$$

### 4.3 Exploitation

Exploitation is among the proposed RL-based RA approach. It involves leveraging accumulated knowledge to make optimal decisions. In the proposed RL-FCRA, the goal refers to make a decision based on the identified policy to minimize the cumulative reward through the reduction of the cost, energy, execution time and latency values.

In algorithm 1, we detail the RL-FCRA based on Q-learning, as a RL algorithm. This approach has as input a set of IoRT-aware BPs modeled using the plug-in presented in [5]. We have divided each IoRT-aware BP into a set of Single Entry Single Exit (SESE) fragments to perform the RA goal using the Refined Process Structure Tree technique (RPST). Each SESE defines its workflow patterns between tasks (e.g., sequence, parallel, split, etc.).

**Table 6.** Comparison of RL-FCRA compared to the other RA approaches regarding the cost, energy, execution time, and latency

| Measured metrics | FIFO | RLRAM | RL-FCRA |
|---|---|---|---|
| Cost ($) | 86.46 | **44.85** | 45.00 |
| Energy(KJ) | 54.50 | 53.3 | **12.87** |
| Execution time (s) | 1.455 | 1.442 | **1.105** |
| Latency (mi/s) | 0.04 | 0,89 | **0,04** |

Moreover, the proposed RL-FCRA has as input a set of Fog resources $A1$ and Cloud ones $A2$ For a specific SESE, the agent receives the coarse-decision (e.g., Fog, Cloud, Fog&Cloud, in-house). According to this decision, the agent initializes the state $S$ and its actions $A1$ and/or $A2$. Subsequently, it invokes the *Resource Allocation* algorithm (refer to Algorithm 2) to determine the optimal resource allocation for each SESE task. This allocation is contingent upon the identified strategy and the proposed equations for calculating rewards.

## 5 Evaluations and Discussion

The goals of our experimental evaluations are two-fold:

– Examine the scalability of the RL-based RA approach concerning the number of (i) SESE, (ii) IoRT-aware BPs (episodes), and (iii) Fog/Cloud resources.

– Estimate the cost, consumed energy, response time, and latency compared to other commonly RA approaches.

To conduct these experiments, we implemented the suggested approach using the Eclipse tool. It is defined as a free and Java-based development platform that allows developers to implement systems and applications within different fields.

In our experiments, we considered a dataset of IoRT-aware BPs in the agriculture field. These processes were defined in the frame of the PRECIMED project [26] and are developed under the Eclipse Modeling Framework (EMF) using the extended BPMN 2.0 modeler plug-in.

In Figure 3, we give an example of the developed IoRT-aware BPs, where this process presents a smart irrigation management system that aims to boost water-use efficiency and nutrients.

We divided the process into various SESE blocks using the Refined Process Structure Tree (RPST) technique. Each SESE has its requirements regarding the million instructions per second (MIPS), random access memory (RAM), size, and bandwidth (BW) (see Table 3).

In the first experiment, we aim to estimate the required time to select adequate resources for each SESE tasks. In this setting, we measure the response time to allocate a resource regarding the number of process fragments and the number of considered processes.

Considering the results presented in Figures 4 and 5, we notice that the proposed RL-FCRA depends (i) on the considering SESE fragments of each IoRT-aware BP and (ii) the considered episodes (number of the IoRT-aware BPs). Therefore, we observe that the response time is directly proportional to the size of the involved fragments and episodes.

The obtained results are justified by the limited capacities (e.g., MIPS, RAM, etc.) of the selected resource, which can result in a longer response time, where the agent struggles to keep up with the requirements of the SESE fragment.

Moreover, we assess the response time of the RL-FCRA by varying the available resources provided by the Fog and Cloud. In this setting, we consider the process presented in Figure 3, which comprises fourteen SESE fragments.

The considered Fog and Cloud resources are presented in table 4. These resources are generated using the ifogSim. The result of this experimentation is shown in Figure 6. Figure 6 shows that the response time depends on the number of available Fog and/or Cloud resources, where the time increases regarding the considered resources.

This is due to the exploitation phase ensured by the RL agent which should estimate the reward for each resource and then select the suitable one. In the context of the RL-FCRA evaluation, we conducted another experiment that aims to

compare the rewarded cost, energy, execution time, and latency of our proposal with other RA approaches. The considered approaches are described in the following:

– FIFO: The First-In-First-Out strategy that aims to implement an unbiased conflict solver because it neglects properties of work items (SESE) and the state of resources [34].
– RLRAM: The Reinforcement Learning Based Resource Allocation Mechanism (RLRAM) aims to propose an optimal RA to the classic BP by trying to minimize the cost [13].

To be able to make this comparison in relevant conditions, the proposed RL-FCRA, RLRAM, and FIFO approaches were executed using the same input configuration of BPs, Fog, and Cloud features. In this setting, we take into account the process (see Figure 5) as an IoRT-aware BP that grouped fourteen SESE fragments, where each SESE has its characteristics (see table **??**). Moreover, we consider a set of twenty Fog resources ($Fog_i$ where i $\in$ $[1..20]$), and twenty Cloud resources ($Cloud_j$ where j $\in [1..20]$). Table 5 gives some examples of the resource features regarding their mips, ram, up bw, down bw, rate per mips, busy power, and idle power values.

Table 6 shows the results of the set forward experiment. It shows that the RL-FCRA outperforms FIFO and RLRAM regarding the consumed energy during the execution of the IoRT-aware BP instance (see table 6). Moreover, we denote from table 6 that our approach requires 1.10s to achieve the execution of the process while the FIFO requires 1.45s to execute it. Furthermore, the estimated latency for the RL-FCRA is around 0.04s. This value is relatively reduced compared to the RLRAM approach which is estimated to be 0.89s. However, for the cost parameter, we note that the proposed approach is less than FIFO.

The experimental results indicate that the RL-FCRA offers a substantial advantage in optimizing the performance of RA in BPM, which confirms the usefulness of our approach and demonstrates its relevance compared to RLRAM and FIFO approaches.

## 6 Conclusion and Future Work

With the proliferation of Industry 4.0, business managers nowadays seek to benefit from IoT and robotics that bring them enormous change in production systems, especially in shorter lead times, flexibility in manufacturing, etc. This led to IoRT-aware BPs. Nonetheless, executing such a process inside the enterprise may be costly due to the consumed resources, the need for computational capacity, etc.

To close these gaps, the business process outsourcing (BPO) strategy can be carried out to externalize the IoRT-aware BP for Cloud and Fog environments. Towards this objective, we proposed a RL-based approach to achieve an optimal allocation of the Fog and Cloud resources for executing an IoRT-aware BP, where this approach addressed the cost, consumed energy, execution time, and latency.

In future directions, we will address some of the current study limitations such as Fog devices' security and mobility issues. Furthermore, in the future, we aim to enhance our proposal by scheduling the RA for the different process fragments.

## References

1. **Abdulqadir, H. R., Zeebaree, S. R., Shukur, H. M., Sadeeq, M. M., Salim, B. W., Salih, A. A., Kak, S. F. (2021).** A study of moving from cloud computing to fog computing. Qubahan Academic Journal, Vol. 1, No. 2, pp. 60–70. DOI: 10.48161/qaj.v1n2a49.
2. **Costa, B., Bachiega, J., de-Carvalho, L. R., Araujo, A. P. F. (2023).** Orchestration in fog computing: A comprehensive survey. ACM Computing Surveys, Vol. 55, No. 2, pp. 1–34. DOI: 10.1145/3486221.
3. **ElHalawany, B. M., Wu, K., Zaky, A. B. (2020).** Deep learning based resources allocation for internet-of-things deployment underlaying cellular networks. Mobile Networks and Applications, Vol. 25, No. 5, pp. 1833–1841. DOI: 10.1007/s11036-020-01566-8.

4. **Fattouch, N., Lahmar, I. B., Boukadi, K. (2021).** A comprehensive architecture for an IoRT-aware business process outsourcing into fog and cloud computing. Proceedings of the Tunisian Algerian Conference on Applied Computing, Vol. 3067, pp. 164–172.

5. **Fattouch, N., Lahmar, I. B., Boukadi, K. (2022).** Towards A meta-modeling approach for an IoRT-aware business process. Proceedings of the 36th International Conference on Modelling and Simulation (ECMS), European Council for Modeling and Simulation, Vol. 36, No. 1, pp. 29–35.

6. **Fattouch, N., Lahmar, I. B., Rekik, M., Boukadi, K. (2022).** Decision-making approach for an IoRT-aware business process outsourcing. Digital, Vol. 2, No. 4, pp. 520–537. DOI: 10.3390/digital2040028.

7. **Gao, H., Jin, T., Feng, C., Li, C., Chen, Q., Kang, C. (2024).** Review of virtual power plant operations: Resource coordination and multidimensional interaction. Applied Energy, Vol. 357, pp. 122284. DOI: 10.1016/j.apenergy.2023.122284.

8. **Govindan, K., Agarwal, V., Darbari, J. D., Jha, P. (2019).** An integrated decision making model for the selection of sustainable forward and reverse logistic providers. Annals of Operations Research, Vol. 273, No. 1, pp. 607–650. DOI: 10.1007/s10479-017-2654-5.

9. **Gupta, H., Dastjerdi, A. V., Ghosh, S. K., Buyya, R. (2017).** iFogSim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments. Software: Practice and Experience, Vol. 47, No. 9, pp. 1275–1296. DOI: 10.1002/spe.2509.

10. **Hailu, T., Chebo, A. K. (2024).** Mapping business process outsourcing and innovation towards a future research. Business Process Management Journal, Vol. 30, No. 1, pp. 158–182. DOI: 10.1108/BPMJ-03-2023-0182.

11. **Halima, R. B., Kallel, S., Ahmed-Nacer, M., Gaaloul, W. (2021).** Optimal business process deployment cost in cloud resources. Journal of Supercomputing, Vol. 77, No. 2, pp. 1579–1611.

12. **Havur, G., Cabanillas, C., Polleres, A. (2022).** Benchmarking answer set programming systems for resource allocation in business processes. Expert Systems with Applications, Vol. 205, pp. 117599. DOI: 10.1016/j.eswa.2022.117599.

13. **Huang, Z., van-der-Aalst, W. M. P., Lu, X., Duan, H. (2011).** Reinforcement learning based resource allocation in business process management. Data & Knowledge Engineering, Vol. 70, No. 1, pp. 127–145. DOI: 10.1016/j.datak.2010.09.002.

14. **Hummaida, A. R., Paton, N. W., Sakellariou, R. (2022).** Scalable virtual machine migration using reinforcement learning. Journal of Grid Computing, Vol. 20, No. 2, pp. 15. DOI: 10.1007/s10723-022-09603-4.

15. **Ihde, S., Pufahl, L., Völker, M., Goel, A., Weske, M. (2022).** A framework for modeling and executing task-specific resource allocations in business processes. Computing, Vol. 104, No. 11, pp. 2405–2429. DOI: 10.1007/s00607-022-01093-2.

16. **Kallel, A., Rekik, M., Khemakhem, M. (2020).** IoT-Fog-Cloud based architecture for smart systems: Prototypes of autism and COVID-19 monitoring systems. Software: Practice and Experience, Vol. 51, No. 1, pp. 91–116. DOI: 10.1002/spe.2924.

17. **Kim, S. H., Jeong, J. H., Aridi, A., Jun, B. (2023).** Factors that affect the technological transition of firms toward the industry 4.0 technologies. IEEE Access, Vol. 11, pp. 1694–1707. DOI: 10.1109/ACCESS.2022.3233390.

18. **Koken, B. (2015).** Cloud robotics platforms. Interdisciplinary Description of Complex Systems: INDECS, Vol. 13, No. 1, pp. 26–33. DOI: 10.7906/indecs.13.1.4.

19. **Lakhan, A., Mohammed, M. A., Obaid, O. I., Chakraborty, C., Abdulkareem, K. H., Kadry, S. (2022).** Efficient deep-reinforcement learning aware resource allocation in SDN-enabled fog paradigm. Automated Software Engineering, Vol. 29, No. 1, pp. 1–25. DOI: 10.1007/s10515-021-00318-6.

20. **López-Pintado, O., Dumas, M., Berx, J. (2024).** Discovery, simulation, and optimization of business processes with differentiated resources. Information Systems, Vol. 120, pp. 102289. DOI: 10.1016/j.is.2023.102289.

21. **Masuda, Y., Zimmermann, A., Shirasaka, S., Nakamura, O. (2020).** Internet of robotic things with digital platforms: Digitization of robotics enterprise. Smart Innovation, Systems and Technologies, Springer, Vol. 189, pp. 381–391. DOI: 10.1007/978-981-15-5784-2_31.

22. **Metzger, A., Kley, T., Palm, A. (2020).** Triggering proactive business process adaptations via online reinforcement learning. Proceedings of the 18th Business Process Management Conference (BPM), Springer, Vol. 12168, pp. 273–290. DOI: 10.1007/978-3-030-58666-9_16.

23. **Neubauer, T. R., da-Silva, V. F., Fantinato, M., Peres, S. M. (2022).** Resource allocation optimization in business processes supported by reinforcement learning and process mining. Lecture Notes in Computer Science, Vol. 13653, pp. 580–595. DOI: 10.1007/978-3-031-21686-2_40.

24. **OMG, O., Parida, R., Mahapatra, S. (2011).** Business process model and notation (BPMN), version 2.0. Object Management Group, Vol. 1, No. 4, pp. 18.

25. **Palm, A., Metzger, A., Pohl, K. (2020).** Online reinforcement learning for self-adaptive information systems. Proceedings of the 32nd International Conference on Advanced Information Systems Engineering (CAiSE), pp. 169–184. DOI: 10.1007/978-3-030-49435-3_11.

26. **Precimed (2022).** precimed. precimed-prima.org/fr/a-propos-de-precimed/.

27. **Rebmann, A., Rehse, J., Pinter, M., Schnaubelt, M., Daun, K., Fettke, P. (2020).** IoT-based activity recognition for process assistance in human-robot disaster response. Business Process Management Forum BPM, Springer, Vol. 392, pp. 71–87. DOI: 10.1007/978-3-030-58638-6_5.

28. **Rekik, M., Boukadi, K., Ben-Abdallah, H. (2017).** A framework for cloud selection based on business process profile. Lecture Notes in Business Information Processing, Springer, Vol. 299, pp. 97–110. DOI: 10.1007/978-3-319-65930-5_.

29. **Simoens, P., Dragone, M., Saffiotti, A. (2018).** The internet of robotic things: A review of concept, added value and applications. International Journal of Advanced Robotic Systems, Vol. 15, No. 1, pp. 1729881418759424. DOI: 10.1177/1729881418759424.

30. **Sutton, R. S., Barto, A. G. (1999).** Reinforcement learning: An introduction. Robotica, Vol. 17, No. 2, pp. 229–235. DOI: 10.1017/S0263574799271172.

31. **Talaat, F. M. (2022).** Effective prediction and resource allocation method (EPRAM) in fog computing environment for smart healthcare system. Multimedia Tools and Applications, Vol. 81, No. 6, pp. 8235–8258. DOI: 10.1007/s11042-022-12223-5.

32. **Volkova, V. N., Chemenkaya, L. V., Desyatirikova, E. N., Hajali, M., Khodar, A., Osama, A. (2018).** Load balancing in cloud computing. Proceedings of the IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), pp. 387–390. DOI: 10.1109/EIConRus.2018.8317113.

33. **Wang, X., Wang, S., Liang, X., Zhao, D., Huang, J., Xu, X., Dai, B., Miao, Q. (2024).** Deep reinforcement learning: a survey. IEEE Transactions on Neural Networks and Learning Systems, Vol. 35, No. 4,

pp. 5064–5078. DOI: 10.1109/TNNLS.2022.3207346.

**34. Yi, L., Cao, S., Wu, Y. (2021).** Coding schemes and resource allocations for the multi-task coded distributed computation. Proceedings of the IEEE International Conference on Communications Workshops (ICC), IEEE, pp. 1–6. DOI: 10.1109/ICCWorkshops50388.2021.9473518.

**35. Zbikowski, K., Ostapowicz, M., Gawrysiak, P. (2021).** Deep reinforcement learning for resource allocation in business processes. Process Mining Workshops, pp. 177–189. DOI: 10.1007/978-3-031-27815-0_13.

*Article received on 03/05/2024; accepted on 12/06/2024.*
**Corresponding author is Najla Fattouch.*

ISSN 2007-9737

# Complaint Process Management in an Electric Power Company

Juan Carlos Blandón-Andrade[1,*], Alejandro Castaño-Toro[1],
Alejandro Morales-Ríos[2], Nilton Tangarife[2]

[1] Universidad Católica de Pereira,
Programa de Ingeniería de Sistemas y Telecomunicaciones,
Colombia

[2] Empresa de Energía de Pereira,
Colombia

{juanc.blandon, alejandro1.castano, alejandro.morales}@ucp.edu.co, ntangarifer@eep.com.co

**Abstract.** The complaint process is a mechanism for citizen participation that provides the means to submit petitions, complaints, and claims to companies providing goods or services. These appeals arrive in large quantities, must be answered in the times established by law, and are costly to process manually. In this article, we propose a computational method to process the complaints written in natural language in Spanish arriving at the Pereira Electric Power Company in Colombia and then classify the complaints that belong to the area of energy solutions to respond in a faster and more effective way. Natural Language Processing and Machine Learning techniques are used to classify the text to construct the method. It starts with the reception of documents for prediction, performs a preprocessing phase, texts are vectorized, a Recurrent Neural Network is configured and trained, and finally, the prediction of each text is presented. The results show that the method processes and classifies the complaints corresponding to the area of electric power solutions and achieves an accuracy of 94.35%, a precision of 95%, a recall of 94%, an F-measure of 94.49% and 93.77% according to the ROC curve metric. The system was tested preliminarily and then with a more formal test in a real environment. Compared to the evaluation criteria of other approaches, the method shows promising results. It was developed under a Service Oriented Software Architecture (SOA) which allowed deployment on a web server and which helps the company to process real complaints efficiently.

**Keywords.** Complaint process, computational methods, electric power company, machine learning, natural language processing.

## 1 Introduction

The complaint process is the mechanism a citizen has to protest verbally or in writing against an administrative irregularity of an entity or company official [25]. In addition, complaints measure and evaluate whether a product or service is well received by users [19].

It is also important to process these inconformities quickly because they must normally be responded to within the time limits established by law [12]. The Natural language processing (NLP) refers to computational techniques for analyzing texts in order to process applications and task. A task is the text classification where assigning one or more categories to pieces of text [3].

Some works highlight the importance of computational processing from texts to contribute to treating complaints in the scientific literature. Some papers highlight the importance of processing complaints related with citizens' complaints about their quality of life in order to create public policies [14, 16].

Others process complaint letters that are published on the Internet and perform automatic classification from the text according to criteria related to company categories [13].

Others focus of citizen complaints in detecting and dealing with pollution events, mentioning that

ISSN 2007-9737



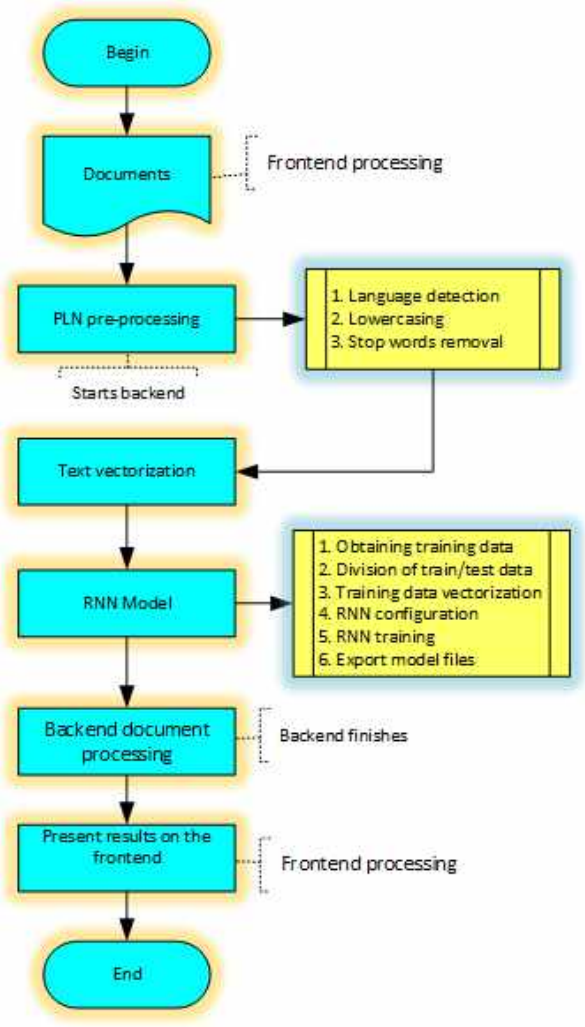



**Fig. 1.** Method proposed

many of these complaints are false and that manual analysis and lack of labeled data further complicate this [9].

Another issue is the treatment of patient complaints in natural language to detect possible signs of adverse drug episodes [27]. Others authors process complaints from the business sector to help strengthen the brands. They mention that it is a great challenge to automatically identify the profusion of online user complaints [23].

In the build sector in China they computationally process the complaints of building occupants,

Text = ['my dog is yellow color','the cat sleeps in my house','my dog is very small']

| | cat | color | dog | house | in | is | my | sleeps | small | the | very | yellow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |

**Fig. 2.** Vectorized text example

thereby seeking to predict the number of thermal complaints to create a facility management strategy, they point out that manual processing is costly and error-prone. Furthermore, they emphasize that automatic classification of complaints is required to improve the efficiency and effectiveness of the process since studies in this area are limited [28, 5].

In the industrial sector feeds a large amount of structured and unstructured data to find information, but a problem arises when operating such a vast amount found in complaints in text format [21]. On the other hand, others authors mention that one way to satisfy a bank's customers is to respond quickly to complaints. However, that number and the categories in which they should be classified are high, so they justify the creation of a computer system to filter them [2].

The Pereira Electric Power Company in Colombia has a high number of users, which is why it receives a significant number of complaints written in natural language in Spanish. Complaints are processed manually, which becomes a very costly process [25]. The specific problem refers to process all the complaints that come to the company, then classify and direction complaints wich belong energy solutions area inside the company.

In this paper, we present a computational method, which uses natural language processing techniques combined with machine learning to process and classify complaints. The method is expected to help the company respond to complaints quickly and effectively. The article is structured as follows: Section 2 discusses related work. Sectión 3 describes the method used to develop the system. Section 4 presents the results. Section 5 presents the discussion. Finally, Section 6 presents the conclusions.

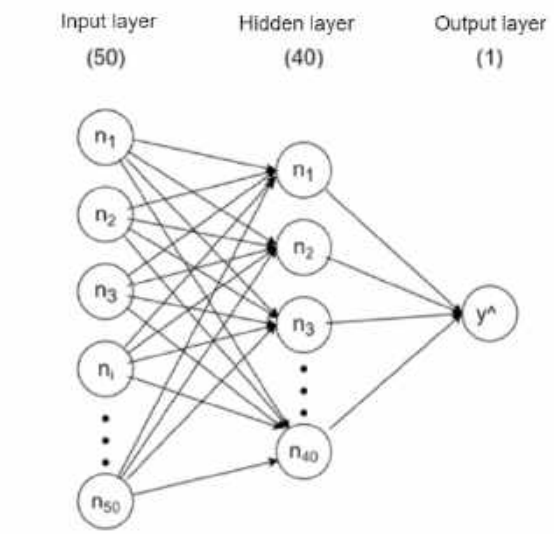



**Fig. 3.** RNN model

## 2 Related Work

Until a few years ago, the computational treatment of natural language was focused on the use of rule-based methods. Then, statistical methods began to be applied, followed by artificial intelligence-based methods used with approaches mainly in Machine Learning, and recently, hybrid methods have been created [8, 6]. Below, some methods for computationally processing complaints are presented.

The methods based on rules are foccus in language rules and their structure. Anggraini et al. [4] developed a system that considered a number of comments from users of a drinking water company, evaluated them using a system of rules, and then established whether it was a positive or negative comment. Farouk et al. [11] took farmers' complaints in Arabic, calculated the semantic similarity, and then tried to solve them.

Usui et al. [27] took complaint data from a Japanese pharmacy system, then designed rules to automatically annotate data taking into account the international code of diseases. Statical methods are based on information, that is, on the distribution of words in the corpus, for which stochastic, probabilistic and statistical methods are used.

Achcar y de Godoy [1] used complaints from users of a telecommunication company about technical service and applied multiple linear regression models and Poisson regression models to detect factors for improvement. Ke y Chen [17] took customer complaints from a gas company, performed preprocessing, segmented the complaint using dictionaries, combined Naive Bayes with the N-Gram model, and analyzed word frequency to perform text classification.

Kim y Lim [18] took user complaints from databases, performed analysis using NLP techniques, built a hierarchy of characteristics, applied, and finally, statistical process control (SPC) analysis for service quality. Artificial intelligence-based methods are focused on learning from data directly, it is important to select good training sets and suitable algorithms.

Fan et al. [10] considered environment-related complaints, created a vocabulary, pre-trained the model, and then trained the TextCNN model for text labeling and classification. Singh y Saha [22] took comments in mixed code, manually annotated the classes from the Product Review dataset, and developed a framework using Graph Attention Network (GAT) by adding some self-service layers.

Thus they managed to detect complaints, classify sentiments, and recognize emotions. Chen et al. [7] took government complaints, applied label correction to refine the labels, and applied Machine Learning algorithms to perform text classification.

Hsu et al. [15] took complaints from preschool children to detect influenza-like illnesses and help detect outbreaks early; they used the BERT algorithm for text classification. Alamsyah et al. [2] took complaints from a bank in Indonesia, then performed preprocessing considering TF-IDF and used Convolutional Neural Network to classify the complaints.

Tong et al. [26] took complaints from a web system, removed negative elements, encoded characters, extracted features, and finally classified the complaints using Convolutional Networks.

**Table 1.** Complaints examples

| Complaint | AREA |
|---|---|
| Usuario solicita un histórico del valor facturado por el concepto de alumbrado público desde enero del 2021 hasta septiembre del 2022, también solicita un detallado mes a mes del valor cancelado por el concepto de alumbrado desde enero del 2021 hasta septiembre del 2022 | Collection |
| Se solicita poda de árbol ubicado en la parte exterior Condominio colinas de la reserva PINARES al lado de poste s0129, dado que se encuentra en conflicto con redes eléctricas | Technical |
| Usuario requiere presentar reclamo por consumo correspondiente al período del 14/12/2021 al 13/01/2022, se verificaron lecturas siendo acordes con las facturadas, se explica que este incremento en el consumo no constituye desviación significativa, pide que se haga revisión al medidor | Billing |
| Usuaria solicita a la empresa una prórroga para presentar ante la empresa los documentos completos para hacer el cambio de matrícula provisional a definitiva | Energy solutions |
| Usuario requiere presentar reclamo por consumo correspondiente al período del 14/12/2021 al 13/01/2022, se verificaron lecturas siendo acordes con las facturadas, se explica que este incremento en el consumo no constituye desviación significativa, pide que se haga revisión al medidor | Billing |
| Usuario requiere presentar reclamo por consumo correspondiente al período del 14/12/2021 al 13/01/2022, se verificaron lecturas siendo acordes con las facturadas, se explica que este incremento en el consumo no constituye desviación significativa, pide que se haga revisión al medidor | Energy solutions |

## 3 Methods and Materials

The system was developed and tested on a computer with an 11th generation Intel Core I5 processor with 6 cores and 2.7 GHz frequency, 16 Gb RAM, and a Windows 11 operating system. The waterfall model was used through the phases of analysis, design, implementation, testing, and deployment to organize the development of the web system, as well as a Service Oriented Software Architecture (SOA) through a connector called Representational State Transfer (REST) [24]. The Frontend was developed with the REACT library using the Javascript programming language.

The Backend was developed with the Flask Framework using the Python programming language using the Keras and Scikit-Learn libraries. As an alternative solution to the textual processing of complaints at Pereira Electric Power Company, this article proposes a computational method that develops several phases presented in Fig.1.

The process began in the Frontend with the reading of the written documents containing the complaints in natural language in Spanish, and to which the prediction was made.

A dynamic string array was created where the texts corresponding to the complaints expressed in the documents in each of the positions were stored one by one. In the end, a dynamic array of texts was obtained with the size of the number of complaints to be processed; this information was sent to the Backend of the system.

### 3.1 Preprocessing

A characteristic of natural language processing software is that it works at the sentence level, so preparation and separation of all the text must be done to obtain well-defined words and sentences

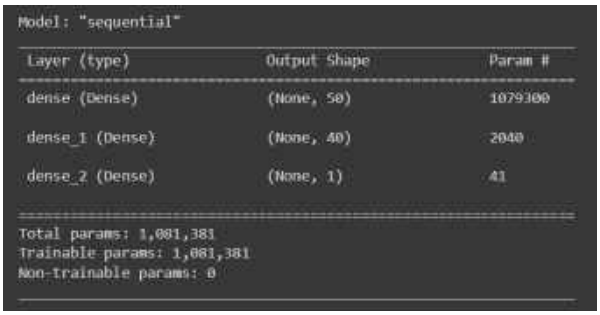



**Fig. 4.** RNN configuration result

for the subsequent phases of the process, which were performed in sequential order.

The Backend method took the text from each position of the array of texts to be predicted and performed the language detection, for which the linguistic probabilities of each sentence were calculated based on orthographic characteristics. If the text was not Spanish, then it was translated using a model of vector representations known as Neuronal Machine Translation (NMT), thus obtaining the Spanish version of the text in each position of the array.

Then the Lowercasing process was performed in all the positions of the array, and the ASCII code was used to find characters in uppercase and change it for its representation in lowercase to normalize the text. Then, frequently-used words, also called empty words in the whole array, i.e., the words that had no meaning, were eliminated.

### 3.2 Text Vectorization

In this phase, a matrix was created where the documents' words were represented uniquely in a column, while the rows were assigned the representation of each text sample of the document (string). Thus, each cell's value is the word count in that particular text sample. Figure 2 presents an example to illustrate in a general way a vectorization matrix.

In the end, obtaining the vectorization matrix of the texts to which the prediction was performed was achieved at this point.

### 3.3 RNN Model

The first part of the model consisted of reading data from an XLSX-extension file with manually-labeled resolved complaints belonging to different areas of the Pereira Electric Power Company. The file had a column containing the text of the complaint and another column indicating whether it belonged to the energy solutions area (1) or not (0).

Then the file's content was divided into two sets: training and testing. For creating the training set, 75% of the complaints contained in the file were taken, and the remaining 25% was used as the test set. Then, the complaint column was taken from the training data file (XLSX file), and another vectorization matrix was created with these data.

#### 3.3.1 Neural Network Configuration

It was decided to use a Recurrent Neural Network (RNN) model to perform the classification process. The configuration was done through a three-layer neuron model; an input layer, a hidden one, and an output one, as shown in Figure 3.

The *input layer* was in charge of receiving the vectorization matrix of the training XLSX file so that the nodes could analyze, classify and share the characteristics to the hidden layer. It was modeled with fifty neurons and a rectifying activation function ReLU, presented by equation 1:

$$f(x) = \max(0, x). \tag{1}$$

The problem involved identifying whether the text belonged to energy solutions area so that it could be mathematically interpreted as a probability ranging between values 0 and 1. The rectifying activation function avoided possible negative data before being sent to the next layer. The *hidden layer* was in charge of receiving the data from the input layer and analyzing them. We were used 40 neurons with a sigmoid-type activation function to achieve this goal, as shown in equation 2:

$$P(t) = \frac{1}{1 + e^{-t}}. \tag{2}$$

The function activates the neuron weights for values between 0 and 1, so it is processed

as a probability when a text arrives. If the resulting value is close to zero, it does not belong to the area, and when it is close to one, it does. Finally, in the *output layer*, being a single-label (yes/no), text classification problem, a single neuron represents the output. Similarly, a sigmoid activation function was used to determine the prediction as a probability.

#### 3.3.2 Neural Network Training

The training of the Recurrent Neural Network was performed, having as input the vectorization matrix built from the labeled training data of the XLSX file. The process was carried out through 20 cycles or periods, where backpropagation was performed in each one.

That is to say, a text was processed, and its prediction was obtained and compared with its expected prediction. Then the cost function was calculated, and based on it, a weight adjustment was performed to improve the affinity of the neuron. The cost function is presented in equation 3:

$$P(t) = \frac{1}{2} \times (y^{\wedge} - y)^2. \tag{3}$$

#### 3.3.3 Export Model Files

After configuring and training the network, we exported the model in an file, a system where data is managed hierarchically and efficiently. The format stores large amounts of numerical, graphic, and text data. It is characterized by the fact that it contains several groups that hold more groups or sets of data within it.

On the other hand, the vectorizer was also exported in a pickle file, where the data were serialized, and the weights of the model were stored. In this way, the two files were ready to be used in the system's backend.

### 3.4 Backend Document Processing

In this section, the two files generated in the training phase were placed in the Backend folder of the built web system. The system, at this point, received a request with the prediction text arrangement, then used the trained model, and finally responded with another array containing the probabilities that each document in the initial array could belong to the energy solutions area of the company.

### 3.5 Presentation of Results in the Frontend

Finally, a file was received from the Backend of the system; it was processed, and the initial documents of the dynamic arrangement with the respective probability of belonging to the energy solutions area of the Pereira Electric Power Company were presented in a user-friendly graphic interface.

## 4 Results

The computational method proposed for processing complaints written in natural language in Spanish at the Pereira Electric Power Company received and classified the texts corresponding to the area of energy solutions, considering different forms of writing. As a result, the system responded in acceptable times and reached an accuracy of 94.35%, a precision of 95%, a recall of 94%, an F-measure of 94.49%, and an ROC curve of 93.77%.

### 4.1 Preprocessing and Vectorization

The system's operation started at the Frontend, where 430 written texts of new complaints were received in natural language in Spanish. These texts were stored in an array and sent to the Backend.

There, the *preprocessing* phase was started, where the language was detected to check that the complaints were in Spanish, then they were standardized to small letters, and lastly, the empty words were removed. In the *vectorization* phase, the strings were converted into a matrix of token counts, where a vectorizer

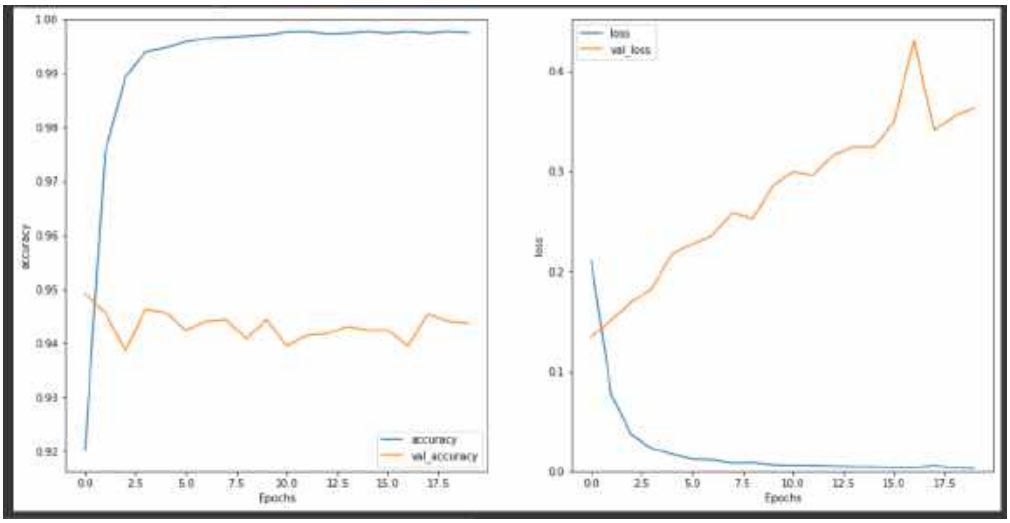



**Fig. 5.** RNN training results

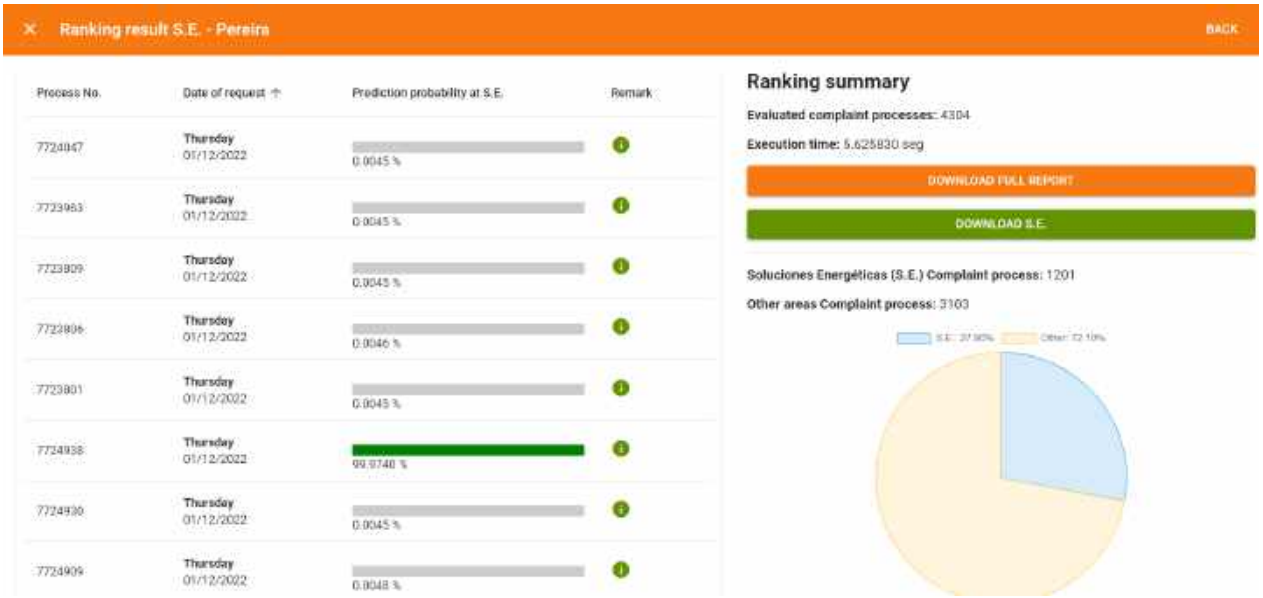



**Fig. 6.** User screen with prediction results

with the vocabulary of all the document tokens for prediction was obtained.

### 4.2 RNN Model

For the implementation phase of the *RNN Model*, we started by loading an file with 12,436 labeled complaints from all areas of the Pereira Electric Power Company, which were the data for this phase. In the Table 1 some texts examples are showing.

Considering those complaints, 2,979 belonged to the energy solutions area, 23.95%, and 9,457, or 76.05%, belonged to the other company areas.

**Table 2.** Comparison of results of the methods

| Model | Precision | Recall | F-measure |
|---|---|---|---|
| **LSTM** | 89.52% | 83.71% | 86.52% |
| **CNN** | 87.99% | 86.93% | 87.46% |
| **SVM** | 89% | 93% | 90.96% |
| **RNN** | 95% | 94% | 94.49% |

After loading the file, we established the training sets with 75% of the data and the test set with 25% of the data. Then the vectorization of the data in this file was performed, and another pickle file was generated to train the system.

Figure 4 shows the *RNN configuration*, for which the 3 layers were created, an input layer with ReLU activation and the hidden and output layers with Sigmoid activation.

In the *training* phase, we used the vectorization of the training excel XLSX file, where 75% of the group was used for training and 25% was used to validate the prediction results. A total of 1,081,381 parameters were trained in a total of 20 epochs. The result of the training process is presented in Figure 5.

After training the network, two files were exported; one with the representation of the model, and a pkl file with the representation of the training data in a vectorization matrix. These two files complete the RNN configuration and training process.

### 4.3 Experimenting with Others Models

Using the data the RNN model we test others machine learning models. So we use Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) and Support Vector Machine (SVM). In the SVM, we apply the bag-of-words (BoW) technique for feature extraction.

This technique allows represents string data in form of finite vectors because it count the ocurrences of a word within the document. Each word is treated as an individual feature, and the model considers the frequency of these words to analyze and classify the text. The results obtained in those experiments are showing in the Table. 2.

### 4.4 Backend and Frontend Processing

The files resulting from the training were placed in the Backend folder of the web system. Then the array of the complaint texts was taken, processed with the system, and resulted in an array with the probability that each document belongs to the energy solutions area of the energy company and is sent to the Frontend. Finally, the Frontend system received the array with the respective prediction probabilities, processed it, and displayed it in a user-friendly graphical interface, as shown in Figure 6.

## 5 Discussion

We agree with the authors that computational methods are necessary to process complaints automatically through various techniques and thus contribute to the resolution of complaints within the time limits established by law. We also agree with the thesis that this contributes to improving company processes.

In this article, we consider the results obtained in ours initial experiments and showed in the table 2, where RNN model obtained the best performance. For this reason, we propose constructing a computational method based on RNN to process the complaints arriving at the Pereira Electric Power Company in Colombia and then classify the complaints corresponding the energy solutions area.

An initial test of the method consisted of manually drafting 30 complaints regarding energy solutions area. Two staff members who normally classify complaints were then asked to rank the drafted complaints employing a survey. On average, the officers identified 30 complaints, while the system identified 27 complaints corresponding to the area of energy solutions, results that are close to the levels of accuracy and recall obtained.

We also drafted 30 complaints from other areas and asked staff members to classify them. They found that none of the complaints belonged to energy solutions, and the system could also identify that none belonged to the energy solutions area. In a more formal test of the method, 430 new complaints were entered from real users, 124 of

**Table 3.** Result of methods in different domains

| Author | Accuracy | Precision | Recall | F-measure | Macro-F1 | Curva Roc |
|---|---|---|---|---|---|---|
| Farouk et al. [11] | – | – | – | 86.7% | – | – |
| Usui et al. [27] | – | 66% | 63% | 64.5% | – | – |
| Hsu et al. [15] | 72.87% | – | – | – | – | – |
| Alamsyah et al. [2] | 85% | – | – | – | – | – |
| Singh and Saha [22] | 72.82% | – | – | – | 71% | – |
| Qurat-ul-ain et al. [21] | 85% | – | – | – | – | – |
| Yance Nanlohy et al. [20] | – | 91.38% | 90.73% | 91.1% | – | – |
| **Our method** | 94.35% | 95% | 94% | 94.49% | – | 93.77% |

which belonged to the energy solutions area of the Pereira Electric Power Company.

The method identified 117 complaints, and the non-identification of the other 7 complaints could be related to the fact that the labeled complaints with which the system was trained could have had typing or spelling errors. In addition to the above, Spanish is one of the most challenging languages to process due to its large number of words (many synonyms), multiple ordering possibilities, semantic differentiation (e.g., verb ser or estar), diatopic varieties (varieties depending on the country), spelling, word length, ambiguity, among others.

The results showed that the computational method processed complaints written in natural language in Spanish and could predict whether or not they belonged to the area of energy solutions in a high percentage. The classification obtained was due to the preprocessing with PLN techniques combined with a 3-layer recurrent neural network that performed well and the extensive training set used.

Some other authors methods for complaint processing achieve exciting results in different application domains. Farouk et al. [11] present a method for processing farmers' complaints in Arabic. The method evaluates semantic similarity and achieves an F-measure of 86.7%. Usui et al. [27] labeled approximately 5,000 complaint documents corresponding to the electronic drug history of a pharmacy in Japan using the international disease code and reported 66% accuracy and 63% recall.

Hsu et al. [15] present a method to process complaints from preschool children to detect influenza and thus help physicians with the diagnosis and act quickly in the event of an outbreak. The authors report an Accuracy of 72.87%. Alamsyah et al. [2] present a method to classify complaints corresponding to 5 units in a bank in Indonesia, they report 85% accuracy, and the authors state that it has not been tested in a real environment.

Singh y Saha [22] propose a method that seeks to exploit complaints from social networks and shopping websites; they obtain a precision of 72.82% and a Macro-F1 of 71%. Qurat-ul-ain et al. [21] propose a method that automatically processes complaints from a web portal, classifies 10,000 complaints into 10 different classes, and obtains an accuracy of 85%.

Yance Nanlohy et al. [20] present a method that receives complaints related to the performance of a government and then classifies them into various categories, obtaining a precision of 91.38% and a recall of 90.73%. A summary of the results of the methods is presented in Table 3.

The results among the methods in the table 3 were obtained from different test data, however, our results are satisfactory. The differences could be due to several factors, for example, the domains they use, the number of documents to process, the size of data in the training sets, and the different technologies used.

In our specific case, using current programming languages, the method could classify the complaints for the energy solutions area of the Pereira Electric Power Company and was tested in a real environment.

## 6 Conclusion and Future Work

In this work, a computational method was proposed to process the different complaints in natural language in Spanish arriving at the Pereira Electric Power Company in Colombia to classify those related to the area of energy solutions. We evaluated different models such as LSTM, CNN and SVM, however, we chose the RNN model because it obtained the best performance.

For the development of the method, PLN tools were used, as well as Machine Learning tools that, employing the proposed methodology and current programming tools, led to the achievement of this work's objective.

The evaluation of the method yielded an accuracy of 94.35%, a precision of 95%, a recall of 94%, and an F-measure of 94.49% and 93.77%, according to the ROC curve metric.

It is essential to clarify that the system was developed under a Service Oriented Software Architecture (SOA), was deployed on a web server, and is currently used in a real environment, contributing to the company's ability to respond to complaints quickly and effectively.

In future work, it is proposed to classify complaints from other company areas by extending the algorithm's scope or using PLN tools combined with other Machine Learning techniques.

## References

1. **Achcar, J. A., de-Godoy, D. M. (2021).** Quality of services: An application with customer complaint data from a telecommunication company. Independent Journal of Management & Production, Vol. 12, No. 4, pp. 928–944. DOI: 10.14807/ijmp.v12i4.1352.
2. **Alamsyah, D. P., Arifin, T., Ramdhani, Y., Hidayat, F. A., Susanti, L. (2022).** Classification of customer complaints: TF-IDF approaches. Proceedings of the 2nd International Conference on Intelligent Technologies, pp. 1–5. DOI: 10.1109/CONIT55038.2022.9848056.
3. **Ameer, I., Sidorov, G., Gómez-Adorno, H., Nawab, R. M. A. (2022).** Multi-label emotion classification on code-mixed text: Data and methods. IEEE Access, Vol. 10, pp. 8779–8789. DOI: 10.1109/ACCESS.2022.3143819.
4. **Anggraini, A., Kusumaningtyas, E. M., Barakbah, A. R., Fiddin-Al-Islami, M. T. (2020).** Indonesian conjunction rule based sentiment analysis for service complaint regional water utility company surabaya. Proceedings of the International Electronics Symposium, pp. 541–548. DOI: 10.1109/IES50839.2020.9231772.
5. **Assaf, S., Srour, I. (2021).** Using a data driven neural network approach to forecast building occupant complaints. Building and Environment, Vol. 200, pp. 107972. DOI: 10.1016/j.buildenv.2021.107972.
6. **Bird, S., Klein, E., Loper, E. (2009).** Natural language processing with Python: Analyzing text with the natural language toolkit. O'Reilly Media, Inc.
7. **Chen, S., Zhang, Y., Song, B., Du, X., Guizani, M. (2022).** An intelligent government complaint prediction approach. Big Data Research, Vol. 30, pp. 100336. DOI: 10.1016/j.bdr.2022.100336.

8. **Clark, A., Fox, C., Lappin, S. (2010).** The Handbook of Computational Linguistics and Natural Language Processing. Wiley-Blackwell.
9. **Fan, Q., Han, H., Wu, S. (2022).** Credibility analysis of water environment complaint report based on deep cross domain network. Applied Intelligence, Vol. 52, No. 7, pp. 8134–8146. DOI: 10.1007/s10489-021-02842-0.
10. **Fan, Q., Yang, K., Qiu, C., Wang, Z. (2021).** Environmental complaint text classification scheme combining automatic annotation and TextCNN. Proceedings of the 2021 China Automation Congress, pp. 4731–4736. DOI: 10.1109/CAC53003.2021.9727661.
11. **Farouk, R. A., Khafagy, M. H., Ali, M., Munir, K., Badry, R. M. (2021).** Arabic semantic similarity approach for farmers complaints. International Journal of Advanced Computer Science and Applications, Vol. 12, No. 10, pp. 348–358. DOI: 10.14569/ijacsa.2021.0121038.
12. **Gómez-Torres, M. P., Rojas-Berrío, S., Robayo-Pinzón, Ó. J. (2015).** Identificación de niveles de calidad en el servicio a partir de peticiones, quejas y reclamos, en entidades bancarias de Colombia 2007-2014. Libre Empresa, Vol. 12, No. 2, pp. 11–26. DOI: 10.18041/libemp.2015.v12n2.24201.
13. **HaCohen-Kerner, Y., Dilmon, R., Hone, M., Ben-Basan, M. A. (2019).** Automatic classification of complaint letters according to service provider categories. Information Processing & Management, Vol. 56, No. 6, pp. 102102. DOI: 10.1016/j.ipm.2019.102102.
14. **Hagen, L., Uzuner, Ö., Kotfila, C., Harrison, T. M., Lamanna, D. (2015).** Understanding citizens' direct policy suggestions to the federal government: A natural language processing and topic modeling approach. Proceedings of the 48th Hawaii International Conference on System Sciences, pp. 2134–2143. DOI: 10.1109/HICSS.2015.257.
15. **Hsu, J. H., Weng, T. C., Wu, C. H., Ho, T. S. (2020).** Natural language processing methods for detection of influenza-like illness from chief complaints. Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, pp. 1626–1630.
16. **Jungherr, A., Jürgens, P. (2010).** The political click: Political participation through E-petitions in Germany. Policy & Internet, Vol. 2, No. 4, pp. 131–165. DOI: 10.2202/1944-2866.1084.
17. **Ke, Y., Chen, L. (2021).** Research on text classification and data analysis of complaints by Shanghai gas company. Proceedings of the IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference, Vol. 5, pp. 1626–1630. DOI: 10.1109/IAEAC50856.2021.9390951.
18. **Kim, J., Lim, C. (2021).** Customer complaints monitoring with customer review data analytics: An integrated method of sentiment and statistical process control analyses. Advanced Engineering Informatics, Vol. 49, pp. 101304. DOI: 10.1016/j.aei.2021.101304.
19. **Mistler, M., Schlueter, N., Loewer, M., Rafalczyk, V. (2021).** Methodology for a model-based traceability of requirements from complaints in business networks using e-DeCoDe. Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1255–1259. DOI: 10.1109/IEEM50564.2021.9673054.
20. **Nanlohy, Y. L., Yuniarno, M. E., Nugroho, S. M. (2020).** Classification of public complaint data in SMS complaint using Naive Bayes multinomial method. Proceedings of the 4th International Conference on Vocational Education and Training, pp. 241–246. DOI: 10.1109/ICOVET50258.2020.9229941.
21. **Qurat-Ul-Ain, Shaukat, A., Saif, U. (2022).** NLP based model for classification of complaints: Autonomous and intelligent system. Proceedings of the 2nd International Conference on Digital Futures and Transformative Technologies, pp. 1–6. DOI: 10.1109/ICoDT255437.2022.9787456.

22. **Singh, A., Saha, S. (2022).** GraphIC: A graph-based approach for identifying complaints from code-mixed product reviews. Expert Systems with Applications, Vol. 216, pp. 119444. DOI: 10.1016/j.eswa.2022.119444.

23. **Singh, A., Saha, S., Hasanuzzaman, M., Dey, K. (2022).** Multitask learning for complaint identification and sentiment analysis. Cognitive Computation, Vol. 14, No. 1, pp. 212–227. DOI: 10.1007/s12559-021-09844-7.

24. **Sommerville, I. (2010).** Software engineering. Pearson Educación S.A.

25. **Stauss, B., Seidel, W. (2019).** Complaint processing. Proceedings of the Effective Complaint Management: The Business Case for Customer Satisfaction, Springer International Publishing, pp. 139–159. DOI: 10.1007/978-3-319-98705-7_8.

26. **Tong, X., Wu, B., Wang, S., Lv, J. (2018).** A complaint text classification model based on character-level convolutional network. Proceedings of the IEEE 9th International Conference on Software Engineering and Service Science, pp. 507–511. DOI: 10.1109/ICSESS.2018.8663873.

27. **Usui, M., Aramaki, E., Iwao, T., Wakamiya, S., Sakamoto, T., Mochizuki, M. (2018).** Extraction and standardization of patient complaints from electronic medication histories for pharmacovigilance: Natural language processing analysis in japanese. JMIR Medical Informatics, Vol. 6, No. 3, pp. e11021. DOI: 10.2196/11021.

28. **Zhong, B., Xing, X., Love, P., Wang, X., Luo, H. (2019).** Convolutional neural network: Deep learning-based classification of building quality problems. Advanced Engineering Informatics, Vol. 40, pp. 46–57. DOI: 10.1016/j.aei.2019.02.009.

*Article received on 25/05/2024; accepted on 12/06/2024.*
**Corresponding author is Juan Carlos Blandón Andrade.*

ISSN 2007-9737

# Comprehensive Method for Measuring Randomness in Pseudorandom Generators

Manuel José Maldonado[1,*], José Luciano Maldonado[2]

[1] Universidad de Los Andes,
Programa de Doctorado en Ciencias Aplicadas,
Venezuela

[2] Universidad de Los Andes,
Instituto de Estadística Aplicada y Computación,
Venezuela

{jlmaldonaj, ingenieriamanuelmaldonado}@gmail.com

**Abstract.** A novel method is presented through which a randomness index is used to measure the quality of the sequences produced by any pseudo-random number generator. To obtain this index, several statistics are combined, some known and others proposed in this research. The method is comprehensive in the sense that, to calculate the randomness index, large sets of data sequences delivered by the generator are considered as a whole. Millions of tests were carried out to generate sequences of pseudo-random numbers and the randomness index was calculated for blocks of these sequences. The results obtained show that the proposed procedure to measure randomness is robust and far outperforms the most popular procedures, which makes it ideal for evaluating the behavior of high-performance generators such as those used in cryptography tasks.

**Keywords.** Randomness index, pseudo-random generators, pseudo-random numbers.

## 1 Introduction

It is widely known that there are a variety of methods to measure the randomness of the samples produced by pseudo-random number generators. As an example, some of these methods are: Frequency test, Gap test, Poker test, Run test, Serial test, Coupon collector's test, Permutation test, Maximum of t test, Collision test, Chi-Square test, Kolmogorov–Smirnov test, Autocorrelation test, and the Spectral test, among other methods.

The operating mechanism of the existing methods, as well as their benefits, are mentioned in various sources [2-4, 9, 12].

However, all these methods have the drawback of measuring only some particular aspects of pseudo-random series, that is, in their statistical analysis they do not treat the series as a whole; then, it may be the case that, with some of such methods, many generators succeed in passing randomness tests, but not with all methods, which leads to the assumption that these methods are not always suitable for measuring randomness.

On the other hand, it is clear that the construction of pseudo-random generators implies, among other tasks, dealing with factors such as 1. The periodicity or presentation of cycles, which is unavoidable in most pseudo-random generators, if not in all.

To highlight this situation, Table 1 shows some of the C++ 11 random class generators [9]. Therefore, if you want to develop an application for cryptographic purposes, these generators are not an option 2.

The predetermined production of pseudo-random series, which occurs in all generators of a mathematical nature, in which a seed, also determined, produces a pseudo-random series known in advance. 3. The Pseudo-random generators' speed of performance, in addition to being dependent on considerations of uniformity, security and independence, are also dependent on response time.

**Table 1**. Characteristics of some pseudo-random generators included in C++11 [9]

| Name | Predictability | Perfomance | State | Period |
|---|---|---|---|---|
| Mersenne Twister | Trivial | OK | 2KiB | 2^1937 |
| LOG(minstd) | Trivial | OK | 4-8 byte | <=2^32 |
| LFG | Trivial | Slow | 8-16byte | <=2^32 |
| Knuth | Trivial | Super slow | 1KB | <=2^32 |
| Ranlux | Unknown | Super slow | *120byte | 10^17 |

Table 1 [9] shows that speed can be a weakness, as is the case for two of the engines presented in that table.

For the reasons mentioned, this paper proposes a method with the aim of contributing to the search for more robust randomness indicators, which allow not only the evaluation of the generators, but also lead to the construction of better pseudo-random generators.

This method was given the name Comprehensive Method for Measuring Randomness in Pseudorandom Generators, ***CMMRPG***, and its design is based on the fact that sequences must be available where the generation of these numbers is almost uniform and independent.

Before explaining the CMMRPG, it is necessary to mention that over the years a variety of test protocols have also appeared, each one combining several methods to measure randomness, among these protocols: KNUTH Protocol that combines the Birthday Spacings tests, Collision test, Coupon collector´s test, Frequency test, Gap test, Maximum-of-t test, Permutation test, Poker test, Run test, Serial test and Serial correlation test; DIEHARD Protocol that combines the Binary Rank tests (31x31,32x32 and 6x8 matrices), Birthday Spacings tests, Bitstreams test, Counts-the-1´s test (for streams and for bytes), Craps test, DNA tests, Minimum distance tests, Overlapping permutations test, Overlapping sums test, Overlapping pairs sparse occupancy test, Overlapping-quadruples-sparse-occupancy test, Parking Lot test, Random spheres test, Run test and Squeeze test; DIEHARDER Protocol that combines all test in Diehard plus the GDC test of Marsaglia-Tsang, Generalized minimum distance test, Lagged sums test, the Permutation test, Monobit test, Run test, the Generalized serial test, the Bit distribution and Kolmogorov-Smirnov test; NIST-15 Protocol that combines Approximate entropy test, the Binary Matrix Rank test, Cumulative sums test, Discrete Fourier Transform test, Frequency test, Frequency within a block test, Linear Complexity test, Maurer's universal statistical test, Non overlapping template matching test, Overlapping template matching test, Random excursions variant test and Serial test; HELSINKI Protocol that combines the Autocorrelation, Cluster test, n-block test and Random walk test; ENT Protocol that combines the Arithmetic mean, Chi square test, Entropy test, Monte-Carlo Pi estimation test and Serial correlation test; Crypt-X Protocol which combines the Binary derivative test, Change point test, Frequency test, Linear complexity test, Run test and Sequence complexity test; SPRING Protocol that combines the Collision test, Coupon Collector´s test, Fourier test, Frequency test, Transform test, Gap test, the Ising Model, Maximum-of-t test, Permutation test, Poker test, Random walk test, Run test and Serial test.

The Diehard test protocol and the NIST 800-22 currently stand out, among the existing protocols above. They are described next:

### 1.1 Diehard Test Protocol

The Diehard protocol [10, 17] consists of a set of tests that, for the most part, returns a ***p***-value, which should be uniform on the interval (0,1) if the input sequence contains truly independent random bits. Those ***p***-values are obtained by ***p=F(X)***, where ***F*** is the distribution of the random variable ***X***, often a normal function. But ***F*** is just an asymptotic approximation, for which the fit will be worse on the tails of the function.

Therefore, we should not be surprised that ***p***-values close to 0 or 1 occasionally appear. When a sequence of bits fails conspicuously, ***p***-values of 0 or 1 will be obtained at many places in the sequence. Since many tests are done, it is likely to find some ***p***<0.025 or some ***p***>0.975, which does not mean that the random number generator fails the test at the 0.05 level.

However, the Diehard protocol does not generate a unified criterion that involves all the tests in a single indicator that measures the random quality of a sample.

### 1.2 NIST 800-22 Test Protocol

The NIST 800-22 test protocol [2, 10, 13], is a suite for randomness analysis, generally used for formal certifications. These tests, like those of the Diehard protocol, analyze various aspects of a sequence of pseudo-random numbers, and have the particularity of being focused on the evaluation of pseudo-random generators intended for cryptographic tasks. Like the DIEHARD protocol, it does not generate a unified indicator for the analyzed data series.

This paper emcompass the Introduction, Section 2, that defines the elements that give rise to the generation of pseudo-random numbers by the generator used in this investigation. Section 3, presents the proposal to use the byte to measure the randomness of pseudo-random generators. Section 4 mentions the pseudo-random generator used in this investigation. Section 5 shows, step by step, how the randomness index proposed in this research is constructed. Section 6 explains the tests performed. Section 7 summarizes an analysis of the results. Section 8 shows the most relevant conclusions of this research.

## 2 Precursors

In this research, the term precursors refer to all the physical events that can be used in the creation of random data, since they are events that precede and give rise to these data. And these data, although they can be called random, at the moment in which the mechanism of occurrence of their precursors is known, they pass into the category of pseudo-random.

On the other hand, today's pseudo-random number generators use number seeds as precursors. Being, by definition, these generators, mathematical-statistical models, whether they are of the type of congruential methods or of another type, which means that they are within the algorithms that Von Newman defined as algorithms that are in "sin", because in no way they generate really random data [6].

Similarly, it is known that we are surrounded by situations where the outcome is quite uncertain, and surely the only thing we can do, a priori, is to assign probabilities to each outcome in the state space of these situations. Therefore, any event whose occurrence is determined by models of complexity superior to the comprehension of human knowledge can be considered random in nature.

So, in this context, and as a way of justifying the use of the definition of precursors, we can consider as valid precursors unexpected natural events such as earthquakes, solar flares, hailstorms and others that cannot be predicted with precision, and that hit the different regions of the planet.

Similarly, in the usual computer equipment, we have variables that are provided by the hardware/software of the system, such as the ticks of the clock, the identification number assigned to a process, the system time at the time of executing a routine and the GUIDs (globally unique identifiers) that can be generated alternately with the readings of other temporary variables.

In any case, and regardless of the context, for a precursor to be eligible as such, any of their multiple's states must have a probability of occurrence close to $\boldsymbol{1/N}$, where $\boldsymbol{N}$ is the number of values that cover the state space of the precursor. It is desirable to have as many valid precursors as possible, in order to build a number generator that meets the standards of sufficient randomness. In this research, events generated by the computer are used as precursors.

## 3 Measure Randomness Using Data Bytes

It is known that, in order to measure the randomness of a data sequence produced by a pseudo-random number generator, common methods proceed to statistical analysis of the bits that make up that data sequence [7, 14-15]. In this research, we worked with the byte instead of the bit, the main motivation for using this unit being the fact that it can represent intelligible entities for humans, such as a letter, a number, a special character, etc.

Another advantage of making randomness measurements with bytes is that the analysis is done on 256 different values, and not on two different values as when working with bits. For example, suppose that a random generator, whose output is given in bits, generates 24 consecutive 1s

digits. This fact allows us to assume a bias, or even a serious failure in the pseudo-random production of the generator. In contrast, if the output is in bytes, three consecutive bytes with the value 255 do not raise the same concern as 24 consecutive equal bits. On the other hand, a graphical, or visual, interpretation of the curves formed by the 256 numbers is more reliable, precisely, because they are formed by a greater variety of data.

## 4 Random Number Generator Used

The tests of the proposed ***CMMRPG*** were carried out on pseudo-random sequences produced by a generator, also novel, which is the product of another investigation carried out by the authors of this paper. This generator is not described in this paper because its operation is being evaluated in another scientific journal, however, its high performance can be highlighted, constituted by a great randomness in the numbers produced. For the purposes of this paper, this generator will be called the Cylinder Mechanism.

This generator, if given the precursors of the type ticks of the computer system clock, the identification number assigned to the process by the operating system, the time of executing a routine, the GUIDs, events of pressing keys, from mouse movements and any other precursors that are not under the user's control; it can produce sequences with very high levels of randomness, which is an ability that makes it superior to the more popular generators [16], and well suited for use in cryptography.

## 5 Construction of the Randomness Index, RI

This index, obtained through ***CMMRPG***, is built following both the Independence criterion and the data uniformity criterion [1,10, 11, 12], and it works with the bytes of the sequences as proposed in section 3.

The ***RI*** construction process begins with the calculation of the following statistics: the Intermedian Comparator (defined as part of this investigation), the arithmetic mean, the median, the standard deviation, the autocorrelation, the interquartile coefficients (defined as part of this investigation), the variation coefficient, the concentration coefficient (defined as part of this investigation), the appearance coefficient (defined as part of this investigation), the correlation index (defined as part of this investigation), the interquartile frequency distribution (defined as part of this investigation), Gap Test Randomness Index and the Index provided by the Test of repetitions (REPI) (defined as part of this investigation).

Next, the proposed statistics are defined, differentiating between those that determine the uniformity and those that determine the independence of the data.

### 5.1 Components to Calculate the Uniformity of Frequency Distribution of the Sequences

Data uniformity is addressed with the following components:

**Intermedian Comparator (IC):** This measure is obtained by comparing the sum of the frequencies on both sides of the median. So, if ***m*** is the median, ***LSS*** is the sum of the frequencies on the left side of ***m*** and ***RSS*** is the sum of the frequencies on the right side of ***m***. If ***f(i)*** is the frequency at each element of the range of ***n*** elements, then ***LSS*** and ***RSS*** are determined as:

$$\begin{aligned} LSS &= \sum_{i=1}^{m-1} f(i), \\ RSS &= \sum_{i=m+1}^{n} f(i). \end{aligned} \tag{1}$$

The ***IC*** is defined in the range $\mathbf{0} \leq IC \leq \mathbf{1}$, and is obtained as follows:

$$\text{If RSS} > \text{LSS, then } IC = \frac{LSS}{RSS} \text{ and otherwise } IC = \frac{RSS}{LSS}$$

**Interquartile Comparators (P12, P23 and P34):** These measures consider the frequencies on both sides of the quartiles.

Let now be the following values:

P1: sum of frequencies of the data with a value lower than quartile 1 (***Q1***).

P2: sum of frequencies of the data between quartile 1 and quartile 2 (***Q2***).

P3: sum of frequencies of the data between quartile 2 and quartile 3 (***Q3***).

P4: sum of frequencies of the data with a value greater than quartile 3 (***Q3***).

Namely:

$$P1 = \sum_{i=1}^{Q1-1} f(i) \quad P2 = \sum_{i=Q1}^{Q2-1} f(i)\,,$$
$$P3 = \sum_{i=Q2}^{Q3-1} f(i)\; y\; P4 = \sum_{i=Q3}^{n} f(i)\; LSS = \sum_{i=1}^{m-1} f(i). \tag{2}$$

In the same way that the ***IC*** is calculated, ***P12***, ***P23*** and ***P34*** are calculated, but considering the frequencies of both sides of each quartile:

$$\mathrm{P12} = \mathrm{P2}/\mathrm{P1} \Leftrightarrow \mathrm{P1}>\mathrm{P2},$$
$$\mathrm{P12} = \mathrm{P1}/\mathrm{P2} \Leftrightarrow \mathrm{P1}\leq\mathrm{P2}, \tag{3}$$

$$\mathrm{P23} = \mathrm{P3}/\mathrm{P2} \Leftrightarrow \mathrm{P2}>\mathrm{P3},$$
$$\mathrm{P23} = \mathrm{P2}/\mathrm{P3} \Leftrightarrow \mathrm{P2}\leq\mathrm{P3}, \tag{4}$$

$$\mathrm{P34} = \mathrm{P4}/\mathrm{P3} \Leftrightarrow \mathrm{P3}>\mathrm{P4},$$
$$\mathrm{P34} = \mathrm{P3}/\mathrm{P4} \Leftrightarrow \mathrm{P3}\leq\mathrm{P4}. \tag{5}$$

**Concentration Coefficient (CC):** Measures how clustered the data is around the mean. For which the Coefficient of Variation (***CV***) is used:

$$\boldsymbol{CV} = \frac{Standard\ deviation}{absolute\ value\ of\ the\ arithmetic\ mean}. \tag{6}$$

And the ***CC*** is defined as ***CC = 1 – CV***.

**Appearance Coefficient (AC):** This coefficient is established to search for abnormalities in the distribution of the data, which the other coefficients mentioned above cannot detect. For example, that a number never appears, rarely or many times. That is, that its appearance value is below the mean minus ***3*** standard deviations or exceeds the mean value plus ***3*** standard deviations. That range is known to pool ***99.7%*** of the data in a normal distribution.

***AC*** is calculated by assigning each frequency value ***f(i)*** a rating ***C(i),*** with ***0≤i ≤255***.

Let be the mean frequency ***X*** and the standard deviation ***σ***, the ***C(i)*** are obtained, for the different values of ***f(i),*** as follows:

If f(i) ≥ 2X ⇒ C(i) = 0

If f(i) > X+3σ ∧ f(i) < 2X ⇒ C(i) = (2X–f(i))/(X-3σ)

If f(i) ≥ X-3σ ∧ f(i) ≤ X+3σ ⇒ C(i) = 1

If f(i) ≤ X-3σ ⇒ C(i) = f(i)/ (X - 3σ)

Once all the ***C(i)*** have been calculated, the ***AC*** is obtained as:

$$AC = \prod_{i=0}^{255} C(i)\,. \tag{7}$$

**Interquartile frequency distribution (IFD):** This measure was defined to establish if the frequency, of the values in the range 0-255, is distributed homogeneously on both sides of each quartile. Each quartile has a size of n/4, where n is the total size of the sample. To calculate this measure, first must be obtained the sum of the frequencies, for each value 0-255, within the limits of each quartile.

Now, for each frequency there are four values or quantities that denotes its appearances within each quartile. With these four values must be calculated the mean frequency *F* and the standard deviation. There is, then, a set of pairs *(Fi, σi),* for *0≤ i≤255*.

For each pair ***(Fi, σi),*** the coefficient of variation ***CVi*** is calculated, thus, 256 coefficients of variation are obtained, as shown in Eq. 8, with ***n=255***.

$$\mathrm{CVar} = \{CV_0, CV_1, CV_2, CV_3, \ldots\ldots CV_n\}. \tag{8}$$

Of these 256 coefficients of variation, the one with the highest value is selected:

$$\mathrm{CVmax} = \max\{\mathrm{CVar}\}. \tag{9}$$

Finally, we have:

$$\mathrm{IFD} = 1 - \mathrm{CV_{max.}} \tag{10}$$

### 5.2 Components to Calculate the Independence of the Sequence Data

Data independence is addressed with the following components.

**Autocorrelation (A)** [5, 8]**:** It is well known that autocorrelation is a very useful tool to measure the independence of data. Let's remember its equation:

$$A = \frac{\sum_{i=1}^{N-h}(x_i - X) \times (x_{i+h} - X_h)}{\sqrt{\sum_{i=1}^{N-h}(x_i - X)^2 \times \sum_{j=H}^{h}\left(x_j - X_h\right)^2}}, \tag{11}$$

where:

***N***: It is the size of the data sequence.

***h***: It is the interval of the regressors.

***X***: Arithmetic mean of the data sequence.
***X***i: the i-th data of the sequence.

The ***A*** has a maximum value of ***1.0*** and a minimum value of ***0.0***. Lower values correspond to more randomness of the data sequence and values close to ***1.0*** represent less randomness. This made it necessary to define the Correlation Index ***CI = 1.0 – A***. Which makes the correlation criterion close to ***1*** mean more randomness, to be in line with the uniformity indices of section 5.1.

**Gap Test Randomness Index (*GAPI*):** to calculate this component of value 0 and 1, the Gap Test was used. Digit sequences were used, and once the gap intervals were established, the Kolmogorov-Smirnov test was used [10-11, 18].

The procedure for calculating the GAPI is as follows:

Let $D^* = D_{calculated}$ and $D = D_{reliability}$

As it is known, if $D_{calculated} < D_{reliability,}$ the digits appear randomly ordered, and if $D_{calculated} > D_{reliability}$, the digits are not randomly ordered.

For a given sequence of digits, we record the number of times the gaps of lengths ***0, 1, 2, …*** appear. After noting how often each gap occurs, the observed relative cumulative frequency ***(Sx)*** is compared to the theoretical cumulative frequency ***F(x).***

Assuming that the digits are randomly ordered, the relative cumulative frequency distribution is given by:

$$Sn(x) = (m / T), \tag{12}$$

where ***m*** is the frequency of the GAP and ***T***, the Total GAPs. The theoretical cumulative frequency distribution is given by:

$$P(gap\ for\ x) = 0.9^x 0.1\ for\ x = 0,1,2,3. \tag{13}$$

Theoretically, the frequency distribution for randomly ordered digits is given by:

$$P(gap\ for\ digits) = F(x) = 0.1 \times \sum_{n=0}^{x} (0.9)^n = 1\text{-}0.9^{x+1}. \tag{14}$$

Let us see an example. If a level of significance ***Sig = 0.05*** is adopted, and the following sequence of digits is obtained:

2, 9, 3, 1, 6, 3, 0, 4, 6, 3, 2, 8, 7, 0, 8, 1, 3, 1, 8, 3, 6, 0, 7, 9, 6, 1, 3, 4, 8, 6, 3, 4, 9, 1, 4, 2, 8, 1, 0, 5, 5, 9, 2, 3, 1, 4, 0, 5, 8, 8, 9, 8, 3, 9, 9, 3, 3, 5, 9, 1, 1, 5, 3, 6, 8, 4, 7, 7, 9, 6, 0, 4, 0, 6, 0, 5, 7, 3, 1, 5, 9, 5, 4, 0, 1, 4, 6, 0, 0, 5, 4, 6, 2, 4, 8, 4, 2, 0, 5, 4, 4, 1, 0, 2, 0, 5, 4, 1, 3, 7, 5, 3, 3, 1, 6, 7, 1, 0, 2, 9, 6, 7, 0, 1, 7.

The **GAPs** number recorded will be the number of digits analyzed minus the number of different random digits generated (in this case, the digits ***0*** to ***9***, that is, ***10*** digits). Total Gaps is ***T=N–10***, where ***N=125*** is the number of digits analyzed. So, ***T = 125 –10 = 115.***

Then, the longest gap length is checked, and the required intervals are determined. For example, if you have a gap length equal to ***49***, and you want ***10*** intervals, then the first interval will be ***0–4***, the second ***5–9***, the third ***10–14***, etc. Conversely, if only 5 intervals are required, then the first interval would be ***0–9***, the second ***10–19***, the third ***20–29***, the fourth ***30–39***, and the fifth ***40– 49.***

For the example sequence, the largest gap length is ***50***, and that sequence was divided into ***17*** intervals. Subsequently, each of the generated random numbers is analyzed to determine its gap length and obtain the frequency in the generated intervals. Thus, selecting the digit ***7***, its first gap length is ***9***; and it will fall in the interval ***9–11***, then that interval will have its first occurrence.

If the same digit or another digit falls in that same interval, then the second occurrence for this interval would be added; and so on for all intervals. The sum of the occurrences, of all the intervals, is equal to the Total **GAPs** (***T=115***).

Gap test steps:

1. The probability function for the theoretical frequency distribution is specified, given by Eq. 14, using the width of the selected intervals.
2. The observed gaps in a cumulative distribution with the same intervals are tabulated.
3. Calculate ***D*,*** the maximum deviation, in absolute value, between ***F(x)*** and ***Sn(x)*** as in Eq. 15.
4. The critical value ***Dα***, from the Kolmogorov–Smirnov table, is determined for the specific value of ***α*** and the sample size ***N***.
5. If ***D**** calculated is greater than Dα, it is concluded that there is no independence of the data, that is, ***GAPI=0***:

$$D = \text{Max} \mid F(x) - Sn(x) \mid. \tag{15}$$

**Table 2.** Calculation of frequencies and differences for the gap text

| Gap size | OC | FA Sn(x) | Frecuencies Fx(x) | \| Fx(x) - Sn(X) \| |
|---|---|---|---|---|
| **0 - 2** | 27 | 0.234 | 0.271 | 0.037 |
| **3 - 5** | 30 | 0.495 | 0.469 | 0.026 |
| **6 - 8** | 23 | 0.695 | 0.613 | 0.082 * |
| **9 - 11** | 11 | 0.792 | 0.718 | 0.074 |
| **12 - 14** | ..... | ..... | ..... | ..... |

***FA*** (Cumulative Frequency = Sn(x)) is calculated with Eq. 16:

$$\mathrm{FA(x)} = \mathrm{Sn(x)} = (\mathrm{m}\ /\mathrm{T}). \tag{16}$$

For the example sequence:
FA of the interval (0–2) = (27 /115) = 0.234.
FA of the interval (3–5) = (57 /115) = 0.495.
FA of the interval (6–8) = (80 /115) = 0.695.
FA of the interval (9–11) = (91 /115) = 0.792.
FA of the interval (…) = (…) = …
(…) = (...) = ...
Fx are calculated as follows:

$$\mathrm{Fx(X)} = 1 - (0.9)^{x+1}. \tag{17}$$

Fx(0–2) = $1 - (0.9)^{2+1}$ = 0.271
Fx(3–5) = $1 - (0.9)^{5+1}$ = 0.469
Fx(6–8) = $1 - (0.9)^{8+1}$ = 0.613
Fx(9–11) = $1 - (0.9)^{11+1}$ = 0.718
Fx(…) = ...
… = …

The absolute maximum difference between ***FA*** and ***Fx***, was, for this test, ***0.082**** as shown in Table 2. The absolute maximum difference is compared with the reliability difference (***Dα***), which is given by Eq. 18:

$$\mathrm{D}\alpha = \textit{value in Kolmogorov-Smirnov's table} \times \sqrt{T}. \tag{18}$$

In Table 2, a sample of the results of the described example is presented. Where ***FA*** and ***OC*** are the cumulative frequency and the occurrence in each interval, respectively. The rows following the range ***9–11*** are omitted as these are not relevant to explaining the results of this example.

Applying a reliability level ***α=0.95***, there is a significance level ***Sig=0.05***, and as the sample size ***N>35***, then, the value in the Kolmogorov-Smirnov table [18] is ***1.36***, therefore, **Dα(0.95)=0.127**. Since ***D*=0.082 < Dα=0.127***, we conclude that the sequence of digits for this example is random. That is, ***GAPI=1***.

**Index by the Test of repetitions (*REPI*):**

This test, which is done on digits, is designed to determine how many times the same repeated digit appears in partial sequences of ***4, 5, 6*** repetitions, in a large sequence of digits.

For example, it can be expected that in a sequence of ***100*** digits there will be up to ***20*** partial sequences of 5 repetitions of the same digit. Of course, if the sample is evenly distributed, that's not going to happen. When a random wheel of digits ***0-9*** spins, each digit has a ***1/10*** chance of appearing, but that doesn't mean that if it is spun 10 times, every single digit will appear.

It is possible, that in a set of ***40*** spin actions, the digit ***1*** only appears once and the digit ***5*** appears ***4*** times, and so on with the other digits. If the number of experiments is large enough, the occurrences of the digits tend to follow a uniform distribution.

There is a sample with ***N*** digits generated by a pseudo-random engine and these are stored in an array from position ***0*** to position ***N-1***. Position ***1*** is set as the start of the experiment. Here, it is worth asking how likely it is that the digit 2 is equal to the digit 1?

Each digit has a 1/10 chance of appearing. If the test is done in position 0, there will be a probability of 1/10 that the digit in that position is equal to the one in position 1. Therefore, there is a probability ***9/10×9/10 = 81/100*** that the digit in position 1 is a non-repeating digit (i.e. is an isolated digit).

It was observed, when consecutive tests were carried out on ***100,000,000*** digits, that, on average, ***80,999,639*** digits were no repeated contiguously, a value very close to that calculated with Eq. 19.

$$\textit{Isolated digits} = (\textit{amount of data}) \times 81/100. \tag{19}$$

For this case, 100,000,000× (81/100)= 80,999,999. For partial sequences of repetitions of ***2, 3, 4, 5, 6*** or more digits, divide the result of Eq. 19, among ***10*** successively, as shown in Table 3.

A decrease given by a factor of ***10*** is observed as more digits are grouped. This was experimentally confirmed with the results shown in Table 4. This is mathematically logical given that each digit added to a previous grouping has a ***1/10*** chance of appearing.

**Table 3**. Ideal appereances Over 100 million digits

| Appereance | | Expected |
|---|---|---|
| Isolated digits: | Expected[0]: | 80,999,999 |
| Pairs: | Expected [1]: | 8,099,999 |
| Triples: | Expected [2]: | 809,999 |
| Quartrains: | Expected [3]: | 80,999 |
| Quintets: | Expected [4]: | 8,099 |
| Sextets: | Expected [5]: | 809 |
| Septets: | Expected [6]: | 80 |
| Octets: | Expected [7]: | 8 |
| 9 repetitions: | Expected [8]: | 0 |

**Table 4**. Occurrences observed for pseudo-random Cylinder Mechanism and C++ 11 intdistro () generators

| Appereances | Cylinders mechanism | intdistro() C++ |
|---|---|---|
| Isolated digits: | 80,993,170 | 81,006,197 |
| Pairs: | 8,102,948 | 8,097,687 |
| Triples: | 810,370 | 809,514 |
| Quartrains: | 80,946 | 80,989 |
| Quintets: | 8,021 | 8,124 |
| Sextets: | 786 | 787 |
| Septets: | 85 | 75 |
| Octets: | 8 | 7 |
| 9 repetitions: | 0 | 0 |

Table 3 shows the expected or theoretical occurrences over ***100*** million digits. Table 4 shows the occurrences observed over 100 million digits produced by the pseudo-random generators: Cylinder Mechanism and intdistro () of C++11.

An evident similarity of the results produced by the two pseudo-random generators is observed, and also, the similarity between the expected and observed repetitions produced by the two generators.

The differences between the expected and observed repetitions must be subject to a tolerance. That is, a number that measures how much the observed data can be allowed to deviate from the calculated quantities. That tolerance must first be defined for digits without contiguous repetitions.

Let:

**Topdata:** the maximum number of data to be analyzed, for this investigation was ***1,000,000,000*** digits.

**Nr_data:** the number of data of the sample to be analized, that has to be a multiple of ***10***. It is recommended to go from $10^6$ to $10^9$.

**Maxdispersion:** the tolerance that is allowed on the **Topdata** amount of data, equal to ***0.0050*** for one billion digits. That is, between the expected amount and the observed amount there can only be a difference of up to ***50,000*** occurrences in ***1,000*** million digits. The number ***50,000*** is called **Tolerance**.

Let $k = log_{10}$(Topdata) - $log_{10}$(Nr_data), then, for any Nr_data, between $10^6$ and $10^9$:

$$Tolerance(Expected[0]) = \\ = \frac{Topdata \times Maxdispersion}{100 \times 3^k}. \quad (20)$$

Let ***PT*** be the Previous Tolerance and ***ND*** be the Next_dispersion containing the value Maxdispersion/2, the Tolerance(Expected[i]) for ***i>0***, is:

$$Tolerance(Expected[i]) = \frac{PT}{2} - (PT \times ND). \quad (21)$$

Now, let $Mag = log_{10}$(Nr_data).

In order to calculate the subsequent Tolerance(Expected[i]), where ***1 ≤ i < Mag-2***, we have the following algorithm:

*Begin*

*Mag ← log10(Nr_data)*

*Tolerance(Expected[0]] ← eq.20*

*PT ← Tolerance(Expected[0]]*

*ND ← Maxdispersion*

*For i =1 To i < Mag-2 Step 1*

*ND ← ND/2*

*Tolerance(Expected[i] ← eq. 21*

*PT ← Tolerance(Expected[i])*

*End for*

*End*

To qualify the repetition test, proceed as follows:

***4*** test protocols are executed: ***100*** tests on ***1,000,000*** digits, ***100*** on ***10,000,000***, ***100*** on ***100,000,000*** and ***100*** on ***1,000,000,000*** digits, respectively. The total number of failed tests is added and the ***REPI*** index is obtained by eq. 22.

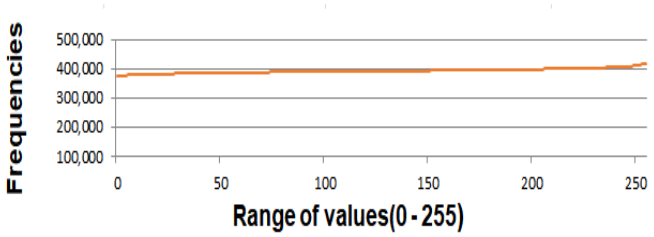



**Fig. 1.** Distribution of average frequencies of the block of 100 runs

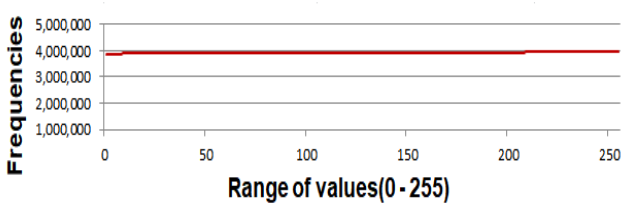



**Fig. 2.** Distribution of average frequencies of the block of 1000 runs

$$REPI = 1 - \frac{\sum_{i=1}^{4} Number\ of\ \ fails\ in\ protocol[i]}{4 \times 100}. \quad (22)$$

### 5.3 Proposed Randomness Index

Finally, the ***IC, P12, P23, P34, DC, AC, IFD, CI, GAPI*** *and* ***REPI*** components are combined to establish the ***RI***:

$$RI = IC \times P12 \times P23 \times P34 \times CC \times AC \times IFD \times CI \times GAPI \times REPI\ . \quad (23)$$

This index has the advantage that any bias, of some importance, in any of the components will be reflected in the result, which is an indication that the randomness of the series being analyzed is compromised in one or several of its components precursors, or that the generator is not an efficient mechanism.

## 6 Convergence Tests of the RI

A wide variety of tests were carried out, of which four large blocks are described.

### 6.1 Block of 100 Runs: Each Run with a Pseudo-Random Sequence of one Million Data

The average statistics obtained for this block are: **Mean:** 392.156,8627451, **Median:** 398.758, **Deviation:** 7.973,9140445, **A:** 0,030035, **LSS:** 50.193.128,000, **RSS:** 49.806.872,000, **IC:** 0,9923046, **P1:** 25.196.104,500, **P2:** 24.797.644,500, **P3:** 24.748.348,500, **P4:** 24.859.144,500, **P12:** 0,984, **P23:** 0,998, **P34:** 0,996, **CC:** 0,979, **AC:** 1, **IFD** = 0,992618 **CI:** 0,9782020, **GAPI =** 1, **REPI** =1 and finally, **RI=**0.92159.

Fig. 1 shows the average frequency distribution of the 256 values present in the pseudo-random sequences used in this run block.

### 6.2 Block of 1000 Runs: Each Run with a Pseudo-Random Sequence of one Million Data

The average statistics obtained for this block are: **Mean:** 392.156,8627451, **Median**: 392.036,600, **Deviation:** 2.027,724, **A:** 0,029985, **IC**: 0,999954968613, **CV:** 0,00517069619, **P12:** 0,9834359, **P23:** 0,9992262, **P34:** 0,9997221, **CC:**

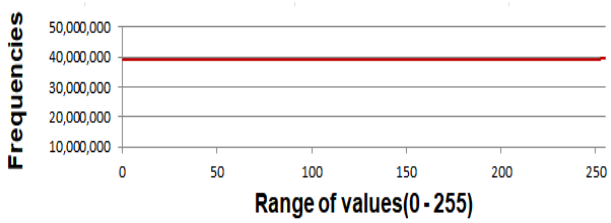



**Fig. 3.** Distribution of average frequencies of the block of 10000 runs

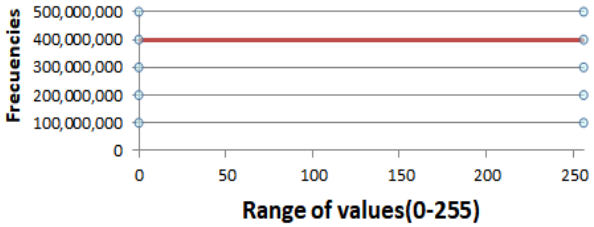



**Fig. 4.** Distribution of average frequencies of the block of 100000 runs

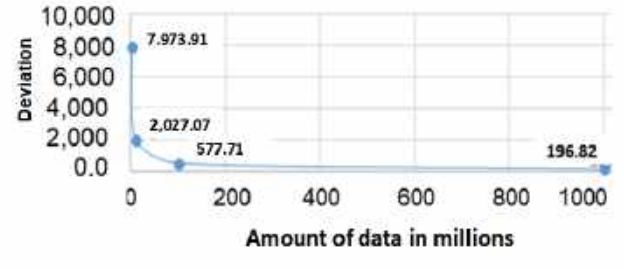



**Fig. 5.** Convergence of the standard deviation to zero

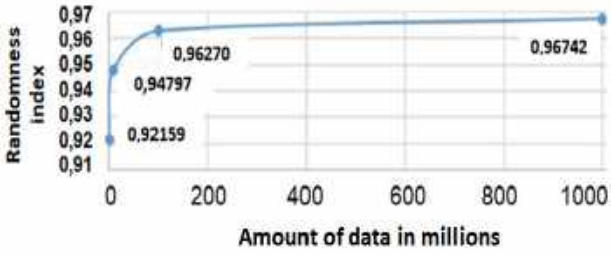



**Fig. 6.** Evolution of the randomness index

0,9948311627, **AC:** 1, **IFD** = 0,992618 **CI:** 0,9772225, **GAPI** =1**, REPI**=1 and finally, **RI=**0,94797.

Fig. 2 shows the average frequency distribution of the 256 values present in the pseudorandom sequences used in this run block.

### 6.3 Block of 10000 Runs: Each Run with a Pseudo-Random Sequence of One Million Data

The average ***RI*** for this block of runs turned out to be 0.96270. Fig. 3 shows the average frequency

distribution of the 256 values present in the pseudo-random sequences used in this run block.

### 6.4 Block of 100000 Runs: Each Run with a Pseudo-Random Sequence of One Million Data

The average *RI* for this block of runs turned out to be *0,96742***.** Fig. 4 shows the average frequency distribution of the 256 values present in the pseudo-random sequences used in this run block.

## 7 Analysis of the Results

In all the tests carried out, an almost uniform distribution of the data was observed, as can be seen in figures 1, 2, 3 and 4.

Fig. 5 shows the convergence of the standard deviation to zero as the increases the volume of data.

Fig. 6 shows a convergence of the ***RI*** to 0.97, which means that the Cylinder Mechanism is an excellent pseudo-random numbers generator, with random quality.

## 8 Conclusions

The ***RI*** is a good measure of the random ability of any generator. In this investigation it was possible to validate its robustness because it worked with the Cylinder Mechanism, which is a high-performance generator and the intdistro() pseudo-random generator of the C++ 11 language.

The tests showed that the Cylinder Mechanism, too, is a robust generator since uniformity and independence of the data could be observed in the range 0-255. The use of precursors is an efficient strategy to achieve the performance capacity of the used pseudo-random number generator. The proposed ***RI*** is very robust against popular randomness measurement methods.

The proposed ***RI*** defines the level of randomness based on the mixture of many statistics, which makes it very reliable. The statistics defined as part of this research are adequate to build the proposed RI.

Popular pseudo-random generators could, in some cases, deliver higher rates, but they being seed-dependent, making them unsuitable for cryptographic uses.

In this investigation, the ***RI*** increased with the amount of data, which verifies the good performance of the Cylinder Mechanism.

The good performance of the Cylinder Mechanism, plus the varied use of precursors, during a test protocol constitute the best option to trigger the generation of pseudo-random numbers with high levels of randomness.

Finally, it can be concluded that the ***CMMRPG*** proposed in this paper constitutes a valid alternative to obtain robust randomness indicators and that, at the same time, allows the construction of robust pseudo-random generators.

## References

1. **Murray, S., Larry, Stephens. (2009).** Theory and problems of statistics. McGraw Hill, Fourth Edition. pp. 61-101.
2. **Marton, K., Suciu, A., Sacarea, C., Cret, O. (2012).** Generation and testing of random numbers for cryptographic applications. Proceedings of the Ramanian Academy, Series A, Vol. 13, No. 4, pp. 368–377.
3. **Ryabko, B. Y., Monarev, V. A. (2005).** Using information theory approach to randomness testing. Journal of Statistical Planning and Inference, Vol. 133, No. 1, pp. 95–110. DOI: 10.1016/j.jspi.2004.02.010.
4. **Calude, C. S. (2013).** Information and randomness: an algorithmic perspective. Springer Science & Business Media.
5. **Downey, R., Hirschfeldt, D. R., Nies, A., Terwijn, S. A. (2006).** Calibrating randomness. Bulletin of Symbolic Logic, Vol. 12, No. 3, pp. 411–491. DOI: 10.2178/bsl/1154698741.
6. **Huang, J. L., Lai, X. J. (2009).** Eliminating ability and correlation of random statistical tests. Information Security and Communications Privacy, Vol. 10, pp. 43–46.
7. **McMillan, M.** (**2020**). Learning C++: generating random numbers the C++11 way. pp 1-6. https://levelup.gitconnected.com.

8. **Demirhan, H., Bitirim, N. (2016).** Statistical testing of cryptographic randomness. İstatistikçiler Dergisi: İstatistik ve Aktüerya, Vol. 9, No. 1, pp. 1–11.
9. **Rukhin, L. (2001).** Testing randomness: a suite of statistical procedures. Theory of Probability & Its Applications, Vol. 45, No. 1, pp. 111–132.
10. **Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Vo, S. (2001).** A statistical test suite for random and pseudorandom number generators for cryptographic applications Gaithersburg, MD, USA: US Department of Commerce, Technology Administration, National Institute of Standards and Technology, Vol. 22, p. 1.
11. **Sulak, F., Doğanaksoy, A., Ege, B., Koçak, O. (2010).** Evaluation of randomness test results for short sequences. In: Carlet, C., Pott, A. (eds) Sequences and Their Applications – SETA 2010. SETA 2010. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg. Vol 6338, DOI: 10.1007/978-3-642-15874-2_27.
12. **Karimovich, G. S., Turakulovich, K. Z., Ubaydullayevna, H. I. (2017).** Computer's source based (Pseudo) random number generation. 2017 International Conference on Information Science and Communications Technologies ICISCTIEEE. pp. 1–6. DOI: 10.1109/ICISCT.2017.8188593.
13. **Valtchanov, B., Fischer, V., Aubert, A., Bernard, F. (2010).** Characterization of randomness sources in ring oscillator-based true random number generators in FPGAs. 13th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems, IEEE, pp. 48–53. DOI: 10.1109/DDECS.2010.5491819.
14. **Panneton, F., L'ecuyer, P. (2005).** On the xorshift random number generators. ACM Transactions on Modeling and Computer Simulation, TOMACS, Vol. 15, No. 4, pp. 346– 361.
15. Pruebas-de-aleatoriedad https//xdoc.mx documents

*Article received on 12/10/2022; accepted on 21/06/2024.*
**Corresponding author is Manuel José Maldonado.*

ISSN 2007-9737

# Non-Intrusive Drowsiness Detection for Accident Prevention Using a Novel CNN

David Hiram Vázquez-Santana, Amadeo José Argüelles-Cruz*

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

{dvazquezs2019, jamadeo}@cic.ipn.mx

**Abstract.** Drowsiness detection problem is not only complex, but also very important for accident prevention. In this paper, we propose a non-intrusive drowsiness detection method using the right eye and mouth. Face detection is performed using HOG + SVM method and facial features are segmented using 8 landmarks obtained by an ensemble of regression trees and classified using a novel convolutional neural network that we call Dozy-Net. Then, drowsiness detection is carried out using three behavioral parameters: PERCLOS, blink frequency, and yawning duration. Two state-of-the-art and one self-constructed dataset were used to train, test, and compare Dozy-Net's performance against other six state-of-the-art convolutional neural networks, being Dozy-Net significantly faster. Drowsiness detection model was tested on a real-life dataset performing 75.8% accuracy and an average speed of 21.51 FPS. Compared to other existing models, our proposal has the advantage of having been tested in conditions similar to those to be expected in a real environment.

**Keywords.** Drowsiness detection; convolutional neural network; behavioral measurement; deep learning; computer vision.

## 1 Introduction

In recent years, our sleep habits have changed negatively due to the amount of time we dedicate to work, hobbies, and transportation. Moreover, the incorporation of new technologies into our lifestyle has also contributed to reducing the time we dedicate to sleep [3].

There are several studies [24, 27, 35, 43] that confirm that between 10% and 13% of the global population frequently suffer from fatigue. Lack of sleep in a minimum of recommended hours [22] drives to several conditions such as drowsiness, irritability, depression, low alertness, anxiety, tension, headache, among others [4, 19, 23, 32]. Drowsiness results in a decreased awareness and both physical and mental performance. It increases the risk of suffering traffic and work accidents [54, 20].

Car accidents tend to be more fatal when caused by drowsiness compared with other causes [13] and employees with sleep problems are 1.62 times more likely to suffer an injury compared to their peers without sleep problems [46].

Social and economic demands have soared in recent years. Nowadays, we spend little time at home and plenty of time at work or in our vehicles, driving from home to work (and vice versa), shopping, going on vacation or visiting other people. In addition, there are several dangerous practices that have become commonly accepted, such as working night shifts or staying up late for leisure after a long workday [15].

Due to our lifestyle, it is important to detect and alert drowsy people to avoid accidents while driving or performing different tasks at work. Nowadays, there are several techniques to detect drowsy people using different techniques [38] based on behavioral parameters, environmental parameters,

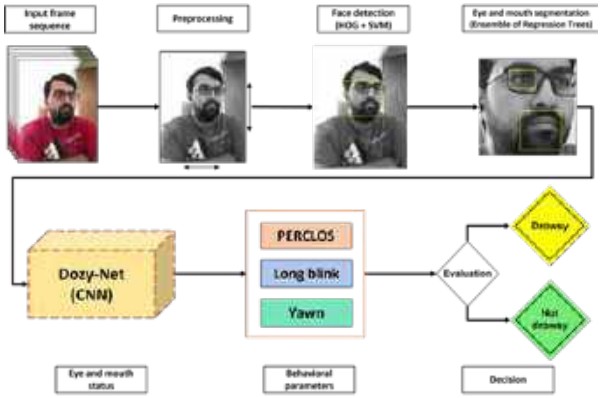



**Fig. 1.** General diagram of the proposed method for drowsiness detection

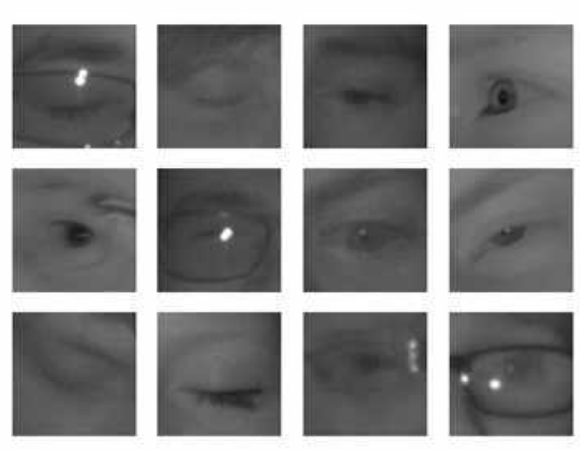


**Fig. 2.** Sample images from the MRL eye dataset

or physiological parameters. This research will focus on drowsiness detection using behavioral parameters obtained from the right eye and mouth of people in different video sequences.

The aim is to classify everyone into one of two classes: drowsy and non-drowsy. This document is organized as follows: in Section 2, related research and state-of-the-art works will be presented.

Section 3 will introduce the methodology used in this research, along with the datasets used, the novel Convolutional Neural Network (CNN) model used for image classification, and the detailed drowsiness detection algorithm.

Section 4 will present the results obtained when detecting drowsy people. Section 5 will discuss the results from this research, and a comparison between state-of-the-art CNN models and our proposal will be presented. Finally, in Section 6, conclusions and future work will be established.

## 2 Related Works

Related work will be divided into three parts: those related to object recognition and tracking, those related to face and facial feature detection, and those describing drowsiness detection methods.

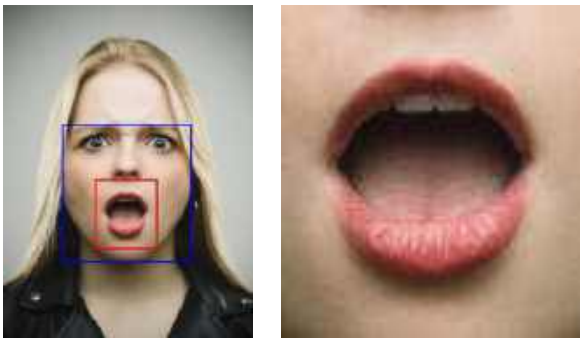
### 2.1 Object Recognition and Tracking

Object recognition is a fundamental task of computer vision and its goal is to recognize particular instances of a certain class in a more complex image. As one of the fundamental tasks of computer vision, object recognition is involved in many other tasks, such as object description or tracking. A vast literature on object detection is available but, in this paper only a few of the most recent works will be mentioned.

The Histogram of Oriented Gradients (HOG) [12] is a feature descriptor which, when combined with a classifier, such as a support vector machine (SVM), can detect objects. The HOG is still widely used despite being a pre-deep learning approach. For example, a vehicle recognition method using a combination of HOG and SVM was recently presented [36] and also a proposal on diabetic retinopathy detection using HOG, and a k-nearest neighbor classifier (KNN) [21].

With the rise of deep learning, it became possible to create more accurate, precise, faster, and more capable models. In this context, the You Only Look Once model (YOLO) is suitable for object tracking due to its high speed on detecting objects [39]. The evolution of YOLO continues to these days and their different versions have been used in a wide variety of applications.

A model for detecting and counting olive fruit flies [29], a vehicle recognition and tracking system in real time [6] and a real time recognition and tracking people during nighttime based on YOLOv3 [30] are some examples.



**(a)** Face of a woman **(b)** Extracted mouth region

**Fig. 3.** Mouth segmentation example

**Fig. 4.** Sample images from the mouths dataset

### 2.2 Face Detection

Face detection is a particular case of object recognition. Its aim is to identify faces in an image regardless of the identity of the person to whom it belongs. Face detection is important to develop different applications, such as a model that detects faces through a hybrid model that combines the Haar cascade classifier and a skin detector for developing an automatic video surveillance [17].

New techniques have been developed to improve facial expression recognition, such as feature descriptors based on geometrical moments [44] or reducing the HOG vector corresponding to the eyes and mouth area using the graph signal processing (GSP) [31].

Thanks to Deep Learning techniques, it has become possible to detect faces even under very difficult conditions, as demonstrated in works where the importance lies in the design of a CNN capable of detecting faces with various alterations that make them difficult to detect, such as blurring, noise or low illumination [7].

### 2.3 Drowsiness Detection

Drowsiness detection methods can be classified into three main categories: techniques based on behavioral parameters [14, 33, 53], environmental parameters [5, 9], or physiological parameters [10, 40, 52].

Environmental parameters-based methods work using data collected by one or more sensors. An example of fatigue detection using this method is the design of a steering wheel that can detect fatigue through eleven features calculated from two driving parameters: steering wheel angle and steering wheel angular velocity.

Both parameters are obtained at a rate of 25 Hz. Steering wheel was tested on ten people (3 women and 7 men) using a driving simulator and the prediction was performed using three models: an SVM, a multilevel ordered logit (MOL) and a neural network. The best accuracy obtained was 74.95% using MOL [9].

The main limitations for using this type of methods are related to the sensors. Installation and cost of the sensors could be high, and external factors can affect the measurements. For example, in driver drowsiness detection, measurements can be affected by pavement or weather conditions.

Physiological fatigue detection methods have been widely used due to the good results obtained with this type of measurements. Electroencephalograms (EEGs) are one example of the physiological measurements and parameters. For example, a work was carried out in the Chinese province of Liaoning, where 16 men took part and EEG signals were obtained through the Emotiv EPOC headset.

The EEGs obtained were processed through a twelve-layer convolutional neural network (CNN) getting an average accuracy of 97.02% [10]. Fatigue detection through physiological parameters is intrusive due to the need of wearable devices, which can cause discomfort in users on everyday activities, making them difficult to implement in vehicles or work scenarios.

Methods based on behavioral parameters have been gaining popularity in recent years due to the capability of non-intrusive drowsiness detection using parameters such as the PERcentage of

**Fig. 5.** Sample images (extracted frames) from the YawDD dataset

**Fig. 6.** Sample images (extracted frames) from the UTA-RLDD dataset

eyelid CLOSure (PERCLOS) over the pupil, facial expressions, blinks, head position, yawns, analysis of physiological patterns over time series [28]. One example of drowsiness detection through PERCLOS and yawning is a model which detects drowsiness if within two minutes a person has yawned at least 3 times or if the PERCLOS value is greater than 0.4.

This method was tested using 10 videos from the YawDD dataset [1] plus one video belonging to one of the authors, achieving an accuracy of 95.91% [49]. Despite the high accuracy percentage, few videos were used to test the model.

Another example is a method that uses the mouth and eye region, PERCLOS and Frequency Of Mouth (FOM) parameters and a Multi-tasking Convolutional Neural Network for drowsiness detection.

This method was tested on the YawDDD [1] and NTHU-DDD [50] datasets achieving an accuracy of 98.72% and 98.91% on YawDD and NTHU-DDD datasets, respectively. Although the performance is good, drowsiness detection is not performed in real time, and the datasets do not take into account truly drowsy people [41].

An example of a work [37] that uses modern recurrent neural networks (RNNs) to classify segments of videos from the eyes and then detects drowsiness with an overall accuracy of 82% with RNNs and 95% with convolutional RNNs. A recent research work in which three descriptors are used to extract information from facial images: the histogram of oriented gradients (HOG); the covariance descriptor (COV); and the local binary pattern were used.

The results provided by each of the descriptors are processed by an SVM and the final decision is reached by merging the individual decisions of each one. The model was tested on the NTHU-DDD [50] dataset which is divided into three sets: training set, evaluation set and test set, scoring an accuracy of 79.84% [33].

In the same way, our proposal focuses on a non-intrusive method that uses several behavioral parameters. Our major contribution in this work is a novel and minimalist CNN, called Dozy-Net, that classifies eye blinking to compute PERCLOS index. that same CNN classifies mouth opening to detect if a person is yawning. As a result, a combination of features allows us to detect drowsy people using only video footage of their faces, trying to avoid further accidents in work or everyday life scenarios.

## 3 Methodology / Proposal

Drowsiness usually exhibits characteristics such as yawning, closed eyes for large periods of time and increased frequency of blinking. This drowsiness detection method is based on three behavioral parameters obtained through video frame analysis: PERCLOS, long blinks and yawns. Considering that the average blink duration is between 174 and 191 milliseconds [34] it is possible to detect them without processing every single frame of the video, thus in this model half of the frames

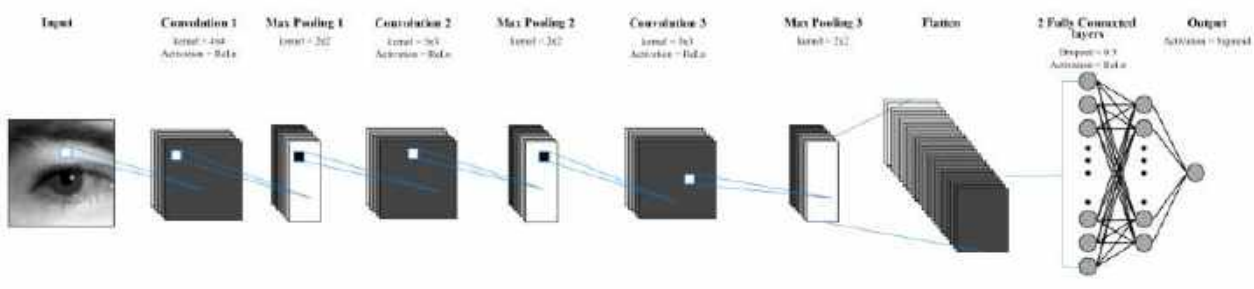



**Fig. 7.** Dozy-Net model

are processed. For example, in a 10-frame sequence $(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10})$, frames $(f_1, f_3, f_5, f_7, f_9)$ will be omitted and frames $(f_2, f_4, f_6, f_8, f_{10})$ will be processed. As shown in 1, the face is detected using HOG + SVM then, the right eye and mouth are segmented using 8 landmarks obtained through an ensemble of regression trees [25].

Both extracted images of facial features are resized and normalized before being classified through our custom designed Dozy-Net, to determine their status. Then, drowsiness is detected through three parameters: Yawning, PERCLOS and long blinks. These behavioral parameters are obtained from the number of continuous frames in which the mouth or eye has been open.

The right eye was tracked since the camera was placed at the right in most of the UTA-RLDD videos [18], making easier to detect and track it. PERCLOS is calculated every 200 frames and since not every frame is processed, it is needed to modify the way that the number of frames is counted and how the PERCLOS value is calculated:

$$\text{PERCLOS} = \frac{f_c \cdot 2}{f_t} \cdot 100. \tag{1}$$

PERCLOS is computed according to equation 1, where $f_c$ and $f_t$ are the amount of closed eye frames and total amount of elapsed frames respectively. In this paper, a person is drowsy if the eye remains closed for 7 or more consecutive processed frames, if PERCLOS is higher than 0.07 or if the mouth remains open for 60 or more consecutive frames. If any of these conditions are present, then the person is considered to be in a vigilant state. As seen in 1, the drowsiness detection system is fully described. Each of the components, including the datasets used for this research, will be described below.

### 3.1 Datasets

#### 3.1.1 Eyes Dataset

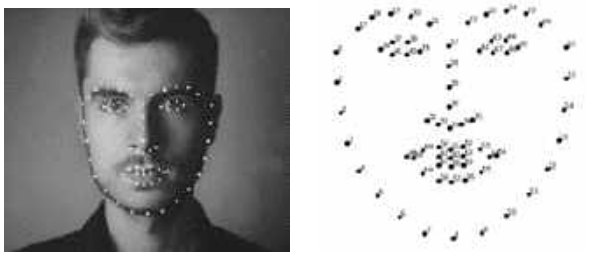
To train our model to classify and detect eye blinking, we have used the MRL Eye Dataset [16]. This dataset contains 84,898 infrared images from 37 persons from which there are 41,945 images of closed eyes and 42,953 of open eyes. 24,001 images are from people that wear glasses. Also, from the total of images, 66,060 do not contain any reflections; 6,129 images contain low reflection levels; and 12,709 contain high reflection levels. From the total of images, 53,630 have poor illumination conditions and the remaining 31,268 have good illumination conditions. MRL Eye Dataset is publicly available at http://mrl.cs.vsb.cz/eyedataset. 2 shows a sample of eye images from the dataset.

#### 3.1.2 Mouths Dataset

There are no publicly available datasets of yawning people that segment the mouth region. Therefore, we build our own dataset from internet-based images from Google, Yandex, and Getty images. We obtained 5,657 images from people with open and closed mouths.


**(a)** Frame resizing **(b)** RGB to grayscale

**Fig. 8.** Image preprocessing



**(a)** Example image **(b)** Enumerated POI

**Fig. 9.** POI extraction

Then, we segmented the region of the mount 3 to build the Mouths dataset which contains 2,805 images from open mouths, and 2,852 images from closed mouths. This dataset is publicly available at https://www.kaggle.com/davidvazquezcic/yawn-dataset. 4 shows an example of the images from the dataset.

#### 3.1.3 Yawn Dataset

We used the YawDD dataset [1], to train our model to detect if a person is yawning or is simply opening its mouth for another casual activity (talking, singing). This dataset contains videos from 57 men and 50 women from different ages and different ethnicities.

Length of videos are between 15 and 40 seconds. The class “yawn” contains 58 videos, and the class “no-yawn” contains 82 videos. In addition, it is important to mention that all videos of YawDD dataset are from people “acting”. Therefore, real-life behavior could vary and that is the reason we only used this dataset to detect yawning and not for detecting drowsy people from the “yawn” class. YawDD is publicly available at https://ieee-dataport.org/open-access/yawdd-yawning-detection-dataset. 5 shows an example of the content of the YawDD dataset.

#### 3.1.4 Drowsiness Dataset

In order to evaluate the effectiveness of our proposal, we used the University of Texas at Arlington Real-Life Drowsiness Dataset (UTA-RLDD) [18]. This dataset provides videos from tired people that guarantee realness in the expressions of the participants. UTA-RLDD contains RGB videos of approximately 10 minutes and were obtained from web cameras and smartphone cameras.

There are videos from 51 men and 9 women of ages from 20 to 59 years old and different ethnicities. There are a total of 180 videos (60 per class) divided in three classes: alert, low vigilant, and drowsy. UTA-RLDD is publicly available at https://sites.google.com/view/utarldd/home. 6 shows an example of the content of the UTA-RLDD dataset.

### 3.2 Network Architecture

There are a variety of algorithms capable of determining the state of selected facial features such as associative memories [2] or associative classifiers [47], in this article, a convolutional neural network is used based on the LeNet classical network, Dozy-Net identifies eye and mouth closure by classifying images of these facial features.

Unlike some famous neural network models belonging to families such as EfficientNet or ResNet which can also achieve good results in facial feature image classification, Dozy-Net is considerably more compact and performs facial feature image classification in a significantly shorter time. The complete Dozy-Net architecture is shown in 7. In CNNs, convolution is used to extract features [26]. The input image is considered as a matrix of size $MxN$ and represented as $W_(m,n)$.

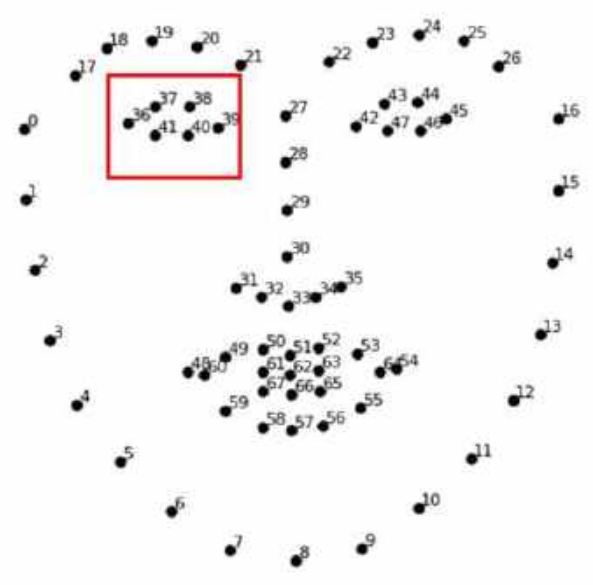


**Fig. 10.** POI used to extract the right eye

This input is convolved with a kernel $k(p,q)$ of size $PxQ$. Dozy-Net is formed by three convolutional layers, the number of layers was chosen empirically aiming to reduce the network size as much as possible but preserving feature-extraction capabilities. The kernel size of the first layer is 4x4, while for the second and third layer the kernel size is 3x3.

The number of kernels for the first, second and third layers are 32, 16 and 8 respectively. A one-pixel stride is applied for all convolutional layers. No padding was used. After each convolutional layer, a max-pooling layer with a kernel size of 2x2 was added. After the max-pooling 3rd layer, data is converted to a 1-dimensional array through flattening.

Feature integration is achieved using two fully connected (FC) layers. The first FC layer has 40 neurons, and the second FC layer has 12 neurons. Finally, classification is performed through a sigmoid activation function neuron. To prevent overfitting, we used the Dropout technique [45] with a dropout probability of 50% in the two FC layers.

### 3.3 Preprocessing and Segmentation

Each frame from the input video is processed according to the following steps:

1. Resize the frame to two different sizes (Display size and learning size, shown in 8a), according to 2.
2. Convert learning size from RGB image format to grayscale image ( 8b):

$$w' = \left[ w \cdot \frac{h'}{h} \right], h' = \left[ h \cdot \frac{w'}{w} \right]. \tag{2}$$

Once the preprocessing is complete, we segmented the face from the grayscale image using HOG+SVM. Moreover, once we obtained the face, we used a regression tree-based algorithm [25] contained in the dlib library for Python programming language to get 68 points of interest (POI) from the extracted face 9.

All images are segmented as shown in 9. Therefore, once the POI is obtained, we extract the face features used in this research to detect drowsiness.

### 3.4 Eye and Blinking Detection

We segmented the right eye from each extracted face through a box that contains the eye, using points 18, 21, 38, and 40 ( 10). Then, we adjust the height of the box 3 to guarantee that the eye is completely extracted:

$$\begin{aligned} y_1' &= y_1 - 8h \cdot 0.11), \\ y_2' &= y_2 - (h \cdot 0.16), \end{aligned} \tag{3}$$

where $y_1$ and $y_2$ correspond to points 38 and 40 respectively and $h$ is the height of the box containing the eye. Once the box is computed, we extract from the input frame the region of the eye, normalize the image and pass it to the Dozy-Net to classify each frame as “open” or “closed”.

We used a threshold of 7 frames (equivalent to approximately 233.8 ms) to determine if a blink was longer than normal [34]. We also use PERCLOS, which is one of the most used behavioral parameters for drowsiness detection [51]. If the PERCLOS value is greater than 0.07, the person is considered to be drowsy.

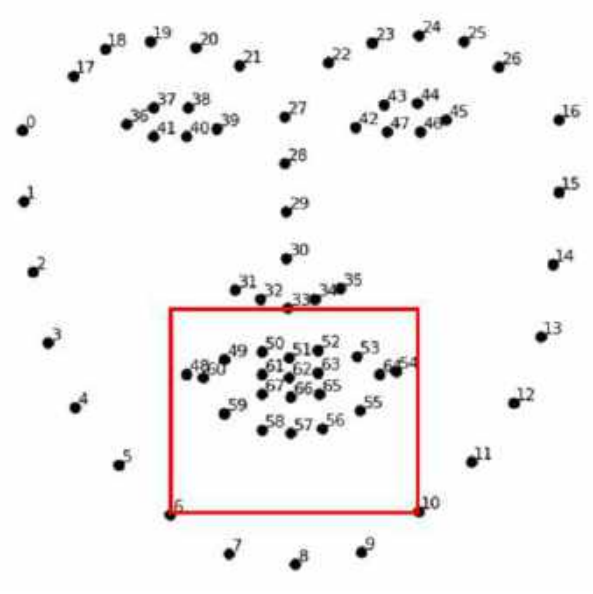


**Fig. 11.** POI used to extract the mouth

### 3.5 Mouth and Yawning Detection

Like the eye detection, we segmented the mouth from the face, using points 6, 10 and 33 11. Once again, we pass the normalized extracted image to the Dozy-Net to classify each frame as “open” or “closed”. We used a threshold of 60 continuous frames (equivalent to approximately 2 seconds) to determine if a mouth opening is associated with the sensory yawning peak, which is the most open phase of a yawn [48].

## 4 Results

### 4.1 Experimental Framework

All experiments presented in this paper were performed on a PC with Intel i7-9750H processor; 8 GB of RAM; 512 SSD storage and an Nvidia GPU GTX 1650 with 4 GB GDDR5 memory. The drowsiness detection model was built in Python 3.7.0 and the main libraries used were Tensorflow-GPU 2.3.0 and Keras 2.4.3 for CNNs implementation, training, and testing; OpenCV 4.5.1 for video and image processing; and dlib 19.21.1 for face landmarks estimation.

Motivated by the good results that VGG-16, ResNet-50, MobileNet-V1, DenseNet-121, NASNet-Mobile and EfficientNet-B2 are able to achieve in image classification [8, 11] (; ; Fulton et al., 2019; Hong et al., 2021; Kundu et al., 2021; Umair et al., 2021; Wang et al., 2020; Yang et al., 2021), we compared the results obtained by the above networks against Dozy-Net.

To ensure a fair comparison, neither transfer learning nor fine-tuning were used, and synaptic weights were randomly initialized. In addition, we used the same learning hyperparameters for all neural networks except for the input size.

All neural networks were trained using a learning rate of 0.001 and binary cross-entropy as the cost function. In addition, we used Adam as an optimization algorithm with parameters $b_1 = 0.9$, $b_2 = 0.999$, and $\epsilon = 1x10^{-7}$, and we set a batch size of 180 to train the models.

### 4.2 Validation Method

MRL Eye Dataset and mouth dataset were used to train and test the facial feature classification models. Therefore, they were divided into three sets following the hold-out validation method: training, validation, and testing.

YawDD was used to determine whether our drowsiness detection model can distinguish yawning from other activities such as singing or talking, and two classes of the UTA-RLDD were used to test our drowsiness detection model. 1 shows the number of patterns per set for each dataset, and the type of data used.

### 4.3 Metrics

Each dataset used in this work is balanced, meaning that there are about the same instances per class. We computed the correctly recognized examples (true positives); the correctly recognized examples that do not belong to each class (true negatives); the examples that were incorrectly assigned to the positive class (false positives); and the examples that were incorrectly assigned to the negative class (false negatives).

**Table 1.** Sets distribution after hold-out validation method

| Dataset | Set | Class 1 | Class 2 | Examples type |
|---|---|---|---|---|
| MRL Eye Dataset | Train | 24,873 | 24,539 | Images |
| | Validation | 16,582 | 16,358 | |
| | Test | 1,497 | 1,497 | |
| Mouths dataset | Train | 1,611 | 1,629 | Images |
| | Validation | 1,074 | 1,086 | |
| | Test | 120 | 137 | |
| YawDD | Test | 82 | 58 | Videos |
| UTA-RLDD | Test | 60 | 60 | Videos |

**Table 2.** Classification results on the eye dataset for all CNN models

| Model | Accuracy | | | Time per epoch (s) |
|---|---|---|---|---|
| | Training | Validation | Test | |
| VGG-16 | 50.32 | 50.33 | 50.00 | 91.82 |
| ResNet50 | 98.81 | 88.37 | **95.22** | 100.53 |
| DenseNet 121 | **98.87** | 85.42 | 94.09 | 75.25 |
| MobileNet | 98.59 | 89.07 | 94.26 | 53.77 |
| NASNet-Mobile | 98.81 | 58.79 | 87.11 | 74.10 |
| EfficientNet B2 | 98.72 | 88.96 | 94.49 | 97.31 |
| **Dozy-Net** | 96.16 | **92.26** | 95.02 | **49.07** |

Therefore, we used accuracy (4) to evaluate the overall effectiveness of different classifiers in a binary classification task [42]:

$$Accuracy = \frac{tp + tn}{tp + fn + fp + tn}. \tag{4}$$

In addition, we also have evaluated the time per epoch that each model takes when training.

### 4.4 Eye Detection Results

The MRL Eye dataset is a balanced set. Therefore, accuracy was used to measure the performance of classification. 2 shows the results from the baselines compared with our Dozy-Net. From 2, we can observe that on validation and testing, our model takes the 1st and 2d place respectively. Moreover, there is a substantial difference in training time compared to our proposal that is twice as fast compared to the best result.

### 4.5 Mouth Detection Results

The own created Mouth dataset is balanced. Therefore, accuracy was also used to measure the performance of classification. 3 shows the results from the baselines compared with our Dozy-Net.

From 3, we can see that our proposal gives us competitive results and it is the fastest among all models. Dozy-Net is up to three times faster than the closer result and up to 2.5 times faster than the best result.

### 4.6 Drowsiness Results

In this section, we present the results obtained by our drowsiness detection model. We use Dozy-Net for facial feature classification in our drowsiness detection model, since it is the fastest.

The first 11,400 frames of each video were analyzed and PERCLOS value was estimated every 200 frames. Through several tests, we found the following judgment conditions for detecting drowsiness:

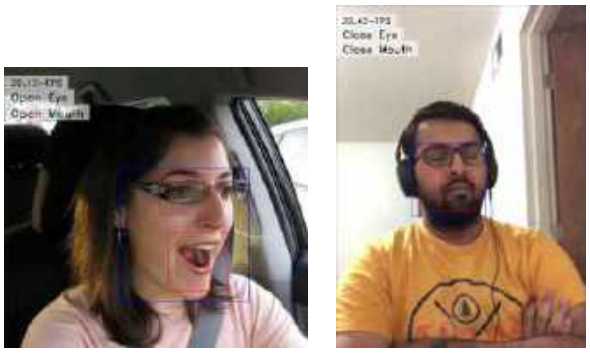
$$\begin{cases} \text{Drowsy}, & (\text{PERCLOS} \geq 0.7) \,||\\ & (F_1 \geq 7\,\text{frames}) \,||\\ & (M_1 \geq 60\,\text{frames})\,, \\ \text{Alert}, & \text{others}, \end{cases} \tag{5}$$

where $F_1$ and $M_1$ are the number of consecutive frames processed from closed eye and open mouth, respectively. 12 shows the result of the classification by Dozy-Net of the facial features segmented through the use of an ensemble of regression trees. We achieved an accuracy of 75.8% in the UTA-RLDD with an average speed of 21.51 FPS.

From the confusion matrix, shown in 13, we can observe that of the 60 cases, 42 people were correctly classified. 18 were misclassified. From the 60 cases where people were fully aware (non-drowsy class), 11 were misclassified and 49 correctly classified using our Dozy-Net. Therefore, the most difficult class for our model was the drowsy class.

**Table 3.** Classification results on the mouths dataset for all CNN models

| Model | Accuracy | | | Time per epoch (s) |
|---|---|---|---|---|
| | Training | Validation | Test | |
| VGG-16 | 50.19 | 50.28 | 53.31 | 14.47 |
| ResNet50 | 96.32 | 87.13 | 96.50 | 14.56 |
| DenseNet 121 | **97.31** | 89.62 | **98.83** | 12.91 |
| MobileNet | 96.34 | 89.83 | 98.44 | 5.15 |
| NASNet-Mobile | 97.10 | 53.43 | 90.66 | 19.48 |
| EfficientNet B2 | 96.59 | 90.59 | 97.67 | 13.06 |
| **Dozy-Net** | 95.02 | **93.62** | 95.33 | **4.60** |



**(a)** YawDDD dataset **(b)** UTA-RLDD

**Fig. 12.** Detection and classification results

### 4.7 Size Comparison

Compared to baseline models, Dozy-Net is tiny and has very few parameters. 4 shows the characteristics of baseline and Dozy-Net models.

## 5 Discussion

Our proposal focuses on the detection of drowsiness from fatigued people. We trained our novel Dozy-Net to classify eyes from being open or closed. From 2, we observed our model obtained the best result of accuracy on the validation set with a 92.26% of accuracy. In addition, our proposal achieved the best result on the test set with 95.05% only after the ResNet50 with 95.22%. However, our model achieved that result from a training two times faster than ResNet50 with only 49.07 seconds per epoch, being the fastest among all CNNs.

Apart from eye detection, our methodology required to detect and classify the mouths of the people from being closed or open. From 3, again we can see that our model achieved the best classification on the validation set with a score of 93.62%.

On the test set we achieved fifth place (95.33%) after the DenseNet 121 (98.83%), MobileNet (98.44%), EfficientNet B2 (97.67%), and the ResNet50 (96.50%). We again obtained the faster training with only 4.60 seconds compared with the models with higher scores DenseNet 121 (12.91s), MobileNet (5.15s), EfficientNet B2 (13.06), and the ResNet50 (14.56).

Our proposal was 2.8 times faster than the best model and 3.1 times faster than the ResNet50. Therefore, in order to obtain the fastest model for fast classification, we performed the evaluation of the UTA-RLDD dataset using the Dozy-Net for eye and mouth classification. We evaluated different conditions for the behavioral parameters mentioned, such as PERCLOS, blink duration, and mouth opening.

We found that best results were obtained using the conditions from equation 5, from which we use a PERCLOS rate greater or equal to 0.07 (7%); for blinking, a number greater than 7 continuous frames; and a mouth opening equal or greater than 60 continuous frames. Only one of the above conditions is needed to classify drowsy people. As a result, we achieved an accuracy of 75.8% on the UTA-RLDD dataset.

On the other hand, from 4 there is no doubt that our Dozy-Net, in addition to being the fastest, has the fewest parameters with only 32.7 thousand compared to the 4.2 million (MobileNet-V1) which is the smallest of the reference models. In other words, our proposal has 128 times fewer parameters than the next in size.

On top of that, the storage size of Dozy-Net is 32 times smaller than the MobileNet-V1 with only 0.5MB compared to 16MB. The smaller storage size and fewer parameters makes our proposal more suitable to obtain faster processing rates and faster FPS in a final real-time application. Therefore, our proposal could be easily implemented on mobile and embedded

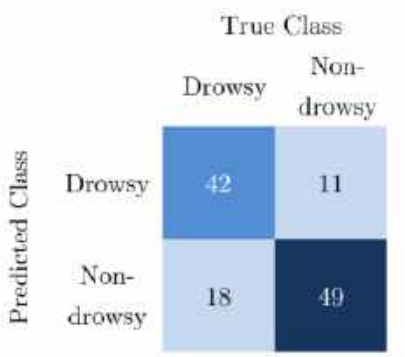



**Fig. 13.** Confusion matrix for our proposed drowsiness detection model

**Table 4.** Number of parameters and size of baseline models and Dozy-Net

| Model | Model size | Parameters |
|---|---|---|
| VGG-16 | 528 MB | 138.3 million |
| ResNet-50 | 98 MB | 25.6 million |
| MobileNet-V1 | 16 MB | 4.2 million |
| DenseNet-121 | 33 MB | 8.1 million |
| NasNet-Mobile | 23 MB | 5.3 million |
| EfficientNet-B2 | 36 MB | 9.1 million |
| **Dozy-Net** | **0.5 MB** | **32.7 thousand** |

devices with a fraction of processing power compared to a full-size GPU.

## 6 Conclusions and Future Work

Drowsiness detection is very important for accident prevention. In this paper we introduce a novel drowsiness detection model based on two facial features and a novel compact convolutional neural network architecture suitable for eye and mouth image classification which we named Dozy-Net.

Dozy-Net proved to be competitive by achieving the shortest classification time among all tested models. It achieved 95.02% and 95.33% accuracy in the eye and mouth image classification test set, respectively, with only 32.7 thousand parameters compared to 25.6 million and 8.1 million parameters of ResNet-50 and DenseNet-121, which were the best performing models in eye and mouth classification, respectively. In addition, the size of Dozy-Net is extremely small being 32 times smaller than MobileNet-V1 and 1056 times smaller than VGG-16.

Finally, our drowsiness detection model combines three behavioral parameters obtained from two facial features. Dozy-Net can detect drowsiness at an average speed of 21.51 FPS using an entry-level GPU. We achieved an accuracy of 75.8% on the UTA-RLDD.

This dataset was selected because, compared to the most commonly used datasets in drowsiness detection, the UTA-RLDD contains information from real drowsy people and thus the exposed facial features are more accurate than synthetic datasets. As a result, our proposal is suitable for real-life scenarios and can positively help prevent and avoid different types of accidents in several real-life scenarios.

The contributions of this work are threefold: (1) the presented convolutional neural network model (Dozy-Net) showed its feasibility to identify critical fatigue features in selected facial features, having 128 times fewer parameters than MobileNet-V1 and 4,229 fewer than VGG-16.; (2) the exploration of thresholds and behavioral parameters suitable for non-intrusive drowsiness detection in real scenarios; and (3) a dataset suitable for yawning detection composed of 5,657 images; 2,805 of open mouths and 2,852 of closed mouths.

Based on the results of the study, we propose to increase the processing speed of Dozy-Net by (1) implementing a reflection removal algorithm on the obtained eye images, (2) binarizing the eye and mouth images through a threshold, and (3) implementing multi-thread processing and a queue data structure for frame storage.

## Acknowledgments

The authors gratefully acknowledge the Instituto Politécnico Nacional (Secretaría Académica, Comisión de Operación y Fomento de Actividades Académicas, Secretaría de Investigación y Posgrado, Centro de Investigación en Computación, and Centro de Innovación y Desarrollo Tecnológico en Cómputo), the Consejo

Nacional de Ciencia y Tecnología (CONACyT), and Sistema Nacional de Investigadores for their economic support to develop this work.

## References

1. **Abtahi, S., Omidyeganeh, M., Shirmohammadi, S., Hariri, B. (2014).** YawDD: A yawning detection dataset. Proceedings of the 5th ACM Multimedia Systems Conference, pp. 24–28. DOI: 10.1145/2557642.2563678.
2. **Acevedo-Mosqueda, M., Yáñez-Márquez, C., López-Yáñez, I. (2007).** Alpha-beta bidirectional associative memories: Theory and applications. Neural Processing Letters, Vol. 26, No. 1, pp. 1–40. DOI: 10.1007/s11063-007-9040-2.
3. **Ahmad-Kamran, M., Naeem-Mannan, M. M., Yung-Jeong, M. (2019).** Drowsiness, fatigue and poor sleep's causes and detection: A comprehensive study. IEEE Access, Vol. 7, pp. 167172–167186. DOI: 10.1109/ACCESS.2019.2951028.
4. **Anaya, F., Abu-Alia, W., Hamoudeh, F., Nazzal, Z., Maraqa, B. (2022).** Epidemiological and clinical characteristics of headache among medical students in palestine: A cross sectional study. BMC Neurology, Vol. 22, No. 1, pp. 1–8. DOI: 10.1186/s12883-021-02526-9.
5. **Arefnezhad, S., Samiee, S., Eichberger, A., Frühwirth, M., Kaufmann, C., Klotz, E. (2020).** Applying deep neural networks for multi-level classification of driver drowsiness using vehicle-based measures. Expert Systems with Applications, Vol. 162, pp. 113778. DOI: 10.1016/j.eswa.2020.113778.
6. **Azimjonov, J., Özmen, A. (2021).** A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways. Advanced Engineering Informatics, Vol. 50, pp. 101393. DOI: 10.1016/j.aei.2021.101393.
7. **Ben-Fredj, H., Bouguezzi, S., Souani, C. (2021).** Face recognition in unconstrained environment with CNN. The Visual Computer, Vol. 37, No. 2, pp. 217–226. DOI: 10.1007/s00371-020-01794-9.
8. **Chaddad, A., Kucharczyk, M. J., Desrosiers, C., Okuwobi, I. P., Katib, Y., Zhang, M., Rathore, S., Sargos, P., Niazi, T. (2020).** Deep radiomic analysis to predict gleason score in prostate cancer. IEEE Access, Vol. 8, pp. 167767–167778. DOI: 10.1109/ACCESS.2020.3023902.
9. **Chai, M., Li, S. W., Sun, W. C., Guo, M. Z., Huang, M. Y. (2019).** Drowsiness monitoring based on steering wheel status. Transportation Research Part D: Transport and Environment, Vol. 66, pp. 95–103. DOI: 10.1016/j.trd.2018.07.007.
10. **Chen, J., Wang, S., He, E., Wang, H., Wang, L. (2021).** Recognizing drowsiness in young men during real driving based on electroencephalography using an end-to-end deep learning approach. Biomedical Signal Processing and Control, Vol. 69, pp. 102792. DOI: 10.1016/j.bspc.2021.102792.
11. **Chen, J., Zhang, D., Suzauddola, M., Zeb, A. (2021).** Identifying crop diseases using attention embedded MobileNet-v2 model. Applied Soft Computing, Vol. 113, pp. 107901. DOI: 10.1016/j.asoc.2021.107901.
12. **Dalal, N., Triggs, B. (2005).** Histograms of oriented gradients for human detection. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, pp. 886–893. DOI: 10.1109/CVPR.2005.177.
13. **Davidović, J., Pešić, D., Lipovac, K., Antić, B. (2020).** The significance of the development of road safety performance indicators related to driver fatigue. Transportation Research Procedia, Vol. 45, pp. 333–342. DOI: 10.1016/j.trpro.2020.03.024.
14. **Elamrani-Abou-Elassad, Z., Mousannif, H., Al-Moatassime, H., Karkouch, A. (2020).** The application of machine learning

techniques for driving behavior analysis: A conceptual framework and a systematic literature review. Engineering Applications of Artificial Intelligence, Vol. 87, pp. 103312. DOI: 10.1016/j.engappai.2019.103312.

15. **Eurofound and International Labour Organization (2019).** Working conditions in a global perspective. op.europa.eu/en/publication-detail/-/publication/19767879-917b-11e9-9369-01aa75ed71a1/language-en. (Accessed on 03/19/2024).
16. **Fusek, R. (2018).** Pupil localization using geodesic distance. Proceedings of the Advances in Visual Computing: 13th International Symposium, ISVC 2018, Springer International Publishing, Cham, pp. 433–444. DOI: 10.1007/978-3-030-03801-4_38.
17. **Gautam, K. S., Thangavel, S. K. (2019).** Video analytics-based intelligent surveillance system for smart buildings. Soft Computing, Vol. 23, No. 8, pp. 2813–2837. DOI: 10.1007/s00500-019-03870-2.
18. **Ghoddoosian, R., Galib, M., Athitsos, V. (2019).** A realistic dataset and baseline temporal model for early drowsiness detection. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society, pp. 178–187. DOI: 10.1109/CVPRW.2019.00027.
19. **Griggs, S., Harper, A., Hickman, R. L. (2022).** A systematic review of sleep deprivation and neurobehavioral function in young adults. Applied Nursing Research, Vol. 63, pp. 151552. DOI: 10.1016/j.apnr.2021.151552.
20. **Hanifah, M. S. A., Ismail, N. (2020).** Fatigue and its associated risk factors: A survey of electronics manufacturing shift workers in malaysia. Fatigue: Biomedicine, Health & Behavior, Vol. 8, No. 1, pp. 49–59. DOI: 10.1080/21641846.2020.1739806.
21. **Hatua, A., Subudhi, B. N., Veerakumar, T., Ghosh, A. (2021).** Early detection of diabetic retinopathy from big data in hadoop framework. Displays, Vol. 70, pp. 102061. DOI: 10.1016/j.displa.2021.102061.
22. **Hirshkowitz, M., Whiton, K., Albert, S. M., Alessi, C., Bruni, O., DonCarlos, L., Hazen, N., Herman, J., Adams-Hillard, P. J., Katz, E. S., Kheirandish-Gozal, L., Neubauer, D. N., O'Donnell, A. E., Ohayon, M., Peever, J., Rawding, R., Sachdeva, R. C., Setters, B., Vitiello, M. V., Ware, J. C. (2015).** National sleep foundation's updated sleep duration recommendations: Final report. Sleep Health, Vol. 1, No. 4, pp. 233–243. DOI: 10.1016/j.sleh.2015.10.004.
23. **Hudson, A. N., Van-Dongen, H. P. A., Honn, K. A. (2020).** Sleep deprivation, vigilant attention, and brain function: A review. Neuropsychopharmacology, Vol. 45, No. 1, pp. 21–30.
24. **Jason, L. A., Evans, M., Brown, M., Porter, N. (2010).** What is fatigue? pathological and nonpathological fatigue. PM&R, Vol. 2, No. 5, pp. 327–331. DOI: 10.1016/j.pmrj.2010.03.028.
25. **Kazemi, V., Sullivan, J. (2014).** One millisecond face alignment with an ensemble of regression trees. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874. DOI: 10.1109/CVPR.2014.241.
26. **Lecun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998).** Gradient-based learning applied to document recognition. Proceedings of the IEEE, Vol. 86, No. 11, pp. 2278–2324. DOI: 10.1109/5.726791.
27. **Loge, J. H., Ekeberg, O., Kaasa, S. (1998).** Fatigue in the general norwegian population: Normative data and associations. Journal of Psychosomatic Research, Vol. 45, No. 1, pp. 53–65. DOI: 10.1016/S0022-3999(97)00291-2.
28. **López-Yáñez, I., Sheremetov, L., Yáñez Márquez, C. (2014).** A novel associative model for time series data mining. Pattern

Recognition Letters, Vol. 41, No. C, pp. 23–33. DOI: 10.1016/j.patrec.2013.11.008.

29. **Mamdouh, N., Khattab, A. (2021).** YOLO-based deep learning framework for olive fruit fly detection and counting. IEEE Access, Vol. 9, pp. 84252–84262. DOI: 10.1109/ACCESS.2021.3088075.
30. **Manssor, S. A. F., Sun, S., Elhassan, M. A. (2021).** Real-time human recognition at night via integrated face and gait recognition technologies. Sensors, Vol. 21, No. 13. DOI: 10.3390/s21134323.
31. **Meena, H. K., Joshi, S. D., Sharma, K. K. (2021).** Facial expression recognition using graph signal processing on hog. IETE Journal of Research, Vol. 67, No. 5, pp. 667–673. DOI: 10.1080/03772063.2019.1565952.
32. **Meerlo, P., Sgoifo, A., Suchecki, D. (2008).** Restricted and disrupted sleep: Effects on autonomic function, neuroendocrine stress systems and stress responsivity. Sleep Medicine Reviews, Vol. 12, No. 3, pp. 197–210. DOI: 10.1016/j.smrv.2007.07.007.
33. **Moujahid, A., Dornaika, F., Arganda-Carreras, I., Reta, J. (2021).** Efficient and compact face descriptor for driver drowsiness detection. Expert Systems with Applications, Vol. 168, pp. 114334. DOI: 10.1016/j.eswa.2020.114334.
34. **Navascues-Cornago, M., Morgan, P. B., Maldonado-Codina, C., Read, M. L. (2020).** Characterisation of blink dynamics using a high-speed infrared imaging system. Ophthalmic and Physiological Optics, Vol. 40, No. 4, pp. 519–528. DOI: 10.1111/opo.12694.
35. **Patel, V., Kirkwood, B., Weiss, H., Pednekar, S., Fernandes, J., Pereira, B., Upadhye, M., Mabey, D. (2005).** Chronic fatigue in developing countries: Population based survey of women in india. BMJ (Clinical research ed.), Vol. 330, No. 7501, pp. 1190. DOI: 10.1136/bmj.38442.636181.E0.
36. **Qu, Z., Chen, Z. (2021).** An intelligent vehicle image segmentation and quality assessment model. Future Generation Computer Systems, Vol. 117, pp. 426–432. DOI: 10.1016/j.future.2020.12.002.
37. **Quddus, A., Zandi, A. S., Prest, L., Comeau, F. J. (2021).** Using long short term memory and convolutional neural networks for driver drowsiness detection. Accident Analysis & Prevention, Vol. 156, pp. 106107. DOI: 10.1016/j.aap.2021.106107.
38. **Ramzan, M., Khan, H. U., Awan, S. M., Ismail, A., Ilyas, M., Mahmood, A. (2019).** A survey on state-of-the-art drowsiness detection techniques. IEEE Access, Vol. 7, pp. 61904–61919. DOI: 10.1109/ACCESS.2019.2914373.
39. **Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016).** You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, pp. 779–788. DOI: 10.1109/CVPR.2016.91.
40. **Rundo, F., Rinella, S., Massimino, S., Coco, M., Fallica, G., Parenti, R., Conoci, S., Perciavalle, V. (2019).** An innovative deep learning algorithm for drowsiness detection from EEG signal. Computation, Vol. 7, No. 1, pp. 13. DOI: 10.3390/computation7010013.
41. **Savaş, B. K., Becerikli, Y. (2020).** Real time driver fatigue detection system based on multi-task ConNN. IEEE Access, Vol. 8, pp. 12491–12498. DOI: 10.1109/ACCESS.2020.2963960.
42. **Sokolova, M., Lapalme, G. (2009).** A systematic analysis of performance measures for classification tasks. Information Processing & Management, Vol. 45, No. 4, pp. 427–437. DOI: 10.1016/j.ipm.2009.03.002.
43. **Son, C. G. (2012).** Review of the prevalence of chronic fatigue worldwide. The Journal of Korean Medicine, Vol. 33, No. 2, pp. 25–33.
44. **Sossa-Azuela, J., Yáñez-Márquez, C., de-León-S, J. (2001).** Computing geometric moments using morphological erosions.

Pattern Recognition, Vol. 34, No. 2, pp. 271–276. DOI: 10.1016/S0031-3203(99)00213-7.

45. **Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014).** Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, Vol. 15, No. 1, pp. 1929–1958.
46. **Uehli, K., Mehta, A. J., Miedinger, D., Hug, K., Schindler, C., Holsboer-Trachsler, E., Leuppi, J. D., Künzli, N. (2014).** Sleep problems and work injuries: A systematic review and meta-analysis. Sleep Medicine Reviews, Vol. 18, No. 1, pp. 61–73. DOI: 10.1016/j.smrv.2013.01.004.
47. **Villuendas-Rey, Y., Rey-Benguría, C., Ferreira-Santiago, Á., Camacho-Nieto, O., Yáñez-Márquez, C. (2017).** The Naïve associative classifier (NAC): A novel, simple, transparent, and accurate classification model evaluated on financial data. Neurocomputing, Vol. 265, pp. 105–115. DOI: 10.1016/j.neucom.2017.03.085.
48. **Walusinski, O., Deputte, B. L. (2004).** Le bâillement: phylogenèse, éthologie, nosogénie. Revue Neurologique, Vol. 160, No. 11, pp. 1011–1021. DOI: 10.1016/S0035-3787(04)71138-8.
49. **Wang, Y., Qu, R. (2021).** Research on driver fatigue state detection method based on deep learning. Proceedings of the 2020 International Conference on Mechanical Automation and Computer Engineering (MACE 2020), Vol. 1744, No. 4, pp. 042242. DOI: 10.1088/1742-6596/1744/4/042242.
50. **Weng, C. H., Lai, Y. H., Lai, S. H. (2017).** Driver drowsiness detection via a hierarchical temporal deep belief network. Computer Vision – ACCV 2016 Workshops, Springer International Publishing, Cham, pp. 117–133. DOI: 10.1007/978-3-319-54526-4_9.
51. **Wierwille, W. W., Wreggit, S. S., Kirn, C. L., Ellsworth, L. A., Fairbanks, R. J. (1994).** Research on vehicle-based driver status/performance monitoring: Development, validation, and refinement of algorithms for detection of driver drowsiness. final report. Research on vehicle-based driver status/performance monitoring : development, validation, and refinement of algorithms for detection of driver drowsiness. Final report, pp. 247.
52. **Wu, E. Q., Deng, P., Qiu, X., Tang, Z., Zhang, W., Zhu, L., Ren, H., Zhou, G., Sheng, R. S. F. (2021).** Detecting fatigue status of pilots based on deep learning network using EEG signals. IEEE Transactions on Cognitive and Developmental Systems, Vol. 13, No. 3, pp. 575–585. DOI: 10.1109/TCDS.2019.2963476.
53. **Yu, J., Park, S., Lee, S., Jeon, M. (2019).** Driver drowsiness detection using condition-adaptive representation learning framework. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, No. 11, pp. 4206–4218. DOI: 10.1109/TITS.2018.2883823.
54. **Zhipeng Peng, H. Z., Wang, Y. (2021).** Work-related factors, fatigue, risky behaviours and traffic accidents among taxi drivers: A comparative analysis among age groups. International Journal of Injury Control and Safety Promotion, Vol. 28, No. 1, pp. 58–67. DOI: 10.1080/17457300.2020.1837885. PMID: 33108968.

*Article received on 16/05/2024; accepted on 01/07/2024.*
**Corresponding author is Amadeo José Argüelles-Cruz.*

ISSN 2007-9737

# Innovation in Mobile Applications with the New Collaborative ASWAN Methodology for Sales Management of Apartments: A Case Study from the Real Estate Sector, Peru

Cinthia Huayana-Salvador[1], Javier Gamboa-Cruzado[1], César Jesús Núñez-Prado[2,*]

[1] Universidad Nacional Mayor de San Marcos, Lima, Peru

[2] Instituto Politécnico Nacional, ESIMEZ, Mexico

{cinthia.huayana, jgamboac}@unmsm.edu.pe, cesar.jnprado@gmail.com

**Abstract.** Companies in the real estate sector often employ ERP systems and commercial management databases (CRM). However, they frequently overlook the integration of mobile technologies in their interactions with clients, which results in decreased efficiency and reduced sales. The primary purpose of this research is to develop a mobile application using an innovative collaborative methodology called ASWAN, aimed at optimizing the sales management of properties in the real estate sector. This study is framed at a descriptive-predictive level and follows a pure experimental design, implementing a software development metamodel known as ASWAN. The results of the study confirm that the mobile application effectively improves sales management indicators, supporting the efficacy of ASWAN in mobile application development. Furthermore, the validity of the proposed hypotheses is statistically demonstrated, pointing out the scarcity of research in the real estate field. The creation of specific management indicators for this sector and the adoption of the ASWAN methodology is suggested due to its design based on the Scrum, RUP, and XP methodologies.

**Keywords**. Mobile application; processes; real estate sector; software development; management indicators; mobile technology.

## 1 Introduction

In recent years, the real estate sector has had to adapt to the new global scenario to continue offering its products and services. Before the pandemic began, apartment sales were initiated and closed in the sales room, with only 5% of placements being made through digital means. Apartment sales declined because sales were not closing due to customer withdrawal and failure to pay the separation fee.

The main objective of the research presented is to develop a mobile application to improve the Management of Apartment Sales in the real estate sector. The application emerges as a technological innovation in the sector, designed with a new methodology and management measurement indicators. Mobile applications and technological innovation are supported by several authors.

According to [14], mobile applications generate innovation as they promote and facilitate the invention and production of new services, products, or processes in the managerial realm. Furthermore, the study [20] points out that new technologies included in Industry 4.0 offer an opportunity to drive technological innovations that can dramatically increase a company's productivity and innovation performance.

Likewise, the authors of the study [15] indicate that current new development methodologies are based on agile principles, applying best practices and iterative phases or processes oriented to maintain flexibility in the face of unexpected changes during project execution.

The authors [3] have written that, by applying behavioral studies to the real estate market,

including perspectives from psychology and sociology through marketing, behavior is governed far beyond simple price discernment. The study [19] states that, in a world constantly stimulated by technological development, companies must acquire structured and organized management control systems. Indeed, companies use new tools daily and make decisions according to available data.

Additionally, the authors [11] clarify that managers must decide what type of knowledge should be generated from each activity to meet their current needs. Thus, depending on the situation, companies must deploy different strategies to acquire tacit and explicit knowledge.

Research [26] indicates that dependence on digital technologies makes it impossible for companies to innovate without technologically complemented process transformations.

Data analysis, AI, and other related technologies offer organizations innovative ways to reinvent processes. According to [21], introducing a mobile app to enhance online commerce and interaction with customers is common in modern organizations. Research has demonstrated its commercial value.

However, few studies have explored the utility of an app that supports, rather than replaces, direct contact with customers. According to [29], social media platforms facilitate the sharing of business information, building relationships with customers, and expressing their emotions.

The development of the Mobile Application with the new ASWAN method focuses on improving the indicators of the commercial area of the real estate sector. Authors [33, 34] demonstrate that the use of Mobile Applications allows reducing times, minimizing deficiencies in processes, and increasing customer satisfaction.

This paper is divided into 6 sections: Section II covers the Theoretical Background, Section III deals with the Research Method, Section IV develops a Case Study, Section V presents Results and Discussions, and finally Section VI Conclusions and Future Research.

## 2 Theoretical Background

### 2.1 Mobile Application

The research is based on mobile applications from a variety where, in the study [14], they foster and simplify the creation and development of new services, products, or processes in large enterprises, providing greater flexibility in management and streamlining the internal procedures of organizations.

The benefits derived go beyond their implementation in business processes, as they enable significant improvements in the quality and efficiency of managerial operations at the heart of organizations.

In [15], the authors suggest that in the development of mobile applications there is a trend towards the use of methodologies with agile principles due to the benefits they provide during the software lifecycle. Software development methodologies seek to cover most of the aspects considered during the execution of a software project with the goal of obtaining quality software products.

### 2.2 Sales Management of Apartments

In the field where the sales management of apartments as established by the authors [3], it is stated that the TCP model was designed to provide explanations of "how" an individual makes behavioral decisions based on a consideration of information; the actual behavior of a person when performing an action is defined by three types of specific beliefs: attitude, subjective norm, and perceived behavioral control.

Therefore, in research [19], it is indicated that other theories provide a more precise approach to investigate digital entrepreneurship. The Theory of Reasoned Action (TRA) highlights intentional behavior, while the Theory of Planned Behavior (TPB) addresses behaviors through a new determining factor called behavioral control.

The authors of the study [11] stated that it is a systematic management of organizational knowledge involving the processes of creation, collection, organization, storage, dissemination, use, and exploitation of knowledge to create

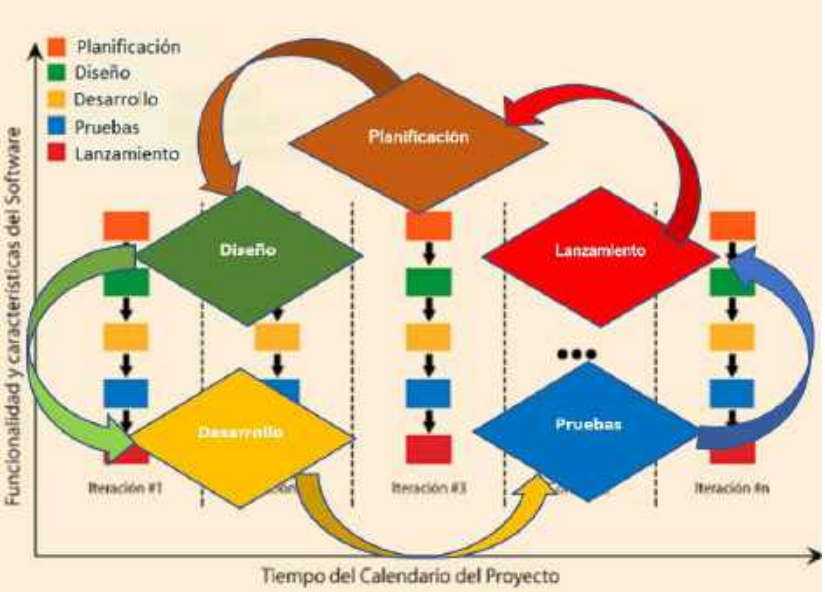



**Fig. 1.** Phases of the new ASWAN methodology

commercial value and generate a competitive advantage.

## 3 Research Method

### 3.1 Generated Collaborative Metamodel: ASWAN METHODOLOGY

It is an agile methodology for software development designed based on the principles of the Scrum, RUP, and Extreme Programming (XP) models, developed in a later section. It consists of five phases as shown in Figure 1.

The phases of the new ASWAN methodology are as follows: Phase 1: Planning, where the product vision is formalized, some user stories are described, risks, time, software requirements are estimated, and the Deliverables Plan is defined. Phase 2: Design, where user stories are defined and use cases are developed, along with the creation of corresponding UML diagrams.

Software requirements are also determined, and the architecture design and system prototype development are carried out. Phase 3: Development, involves coding the program to generate a version of the system, developing the sequence of iterations, the deployment model, and updating risks. Phase 4: Testing, this phase is prior to obtaining the finished version of the system and involves unit testing to detect and correct faults, followed by acceptance testing. Phase 5: Launch, in this phase the owner or user reviews the functionality and quality of the product and receives the developed system and corresponding user manual.

### 3.2 Applied Research Methodology

In this subsection, the specific methodology applied in the study is delved into. Reasons behind the choice of this methodology, how it was adapted to the research context, and the concrete steps that were followed are discussed. Advantages and possible limitations of this approach will also be touched upon. Clarity in this section is crucial for readers to understand the validity and applicability of the results derived.

#### 3.2.1 Operationalization of Variables

Table 1 shows the indicators, their unit of measure, the index, the unit of measure, the instrument for data collection.

**Table 1.** Operationalization of the Dependent Variable

| Indicator | Index | Unit of Measure |
|---|---|---|
| Qualified Prospect Quantity | (0-30) | Leads/ weekly |
| Closed Sales Quantity | (0–20) | Apartments/biweekly |
| Time for Presentation and Demonstration | (15–40) | Minutes/ biweekly |

### 3.2.2 Research Design

This research adopts an experimental design classified under the category of "pure" experiments. It involves comparing the experimental group (Ge), which will receive the stimulus (mobile application) for apartment sales management processes, randomly selected (R), with post-test data yielding O1.

Subsequently, a stimulus (X) based on the collaborative metamodel ASWAN is applied, resulting in post-test data for the control group (Gc), O2, which did not receive the stimulus. This is an applied type of pure experimental design:

RGe X $O_1$.

RGc -- $O_2$.

### 3.2.3 Universe and Sample

The universe for this study was identified as all apartment sales management processes in real estate groups and startup companies across Latin America. These processes are unspecified and indeterminate (N = Indeterminate). The sample consisted of the sales management processes at NERMAR real estate, with a known sufficient sample size of n = 30 for process optimization.

### 3.2.4 Data Collection Procedure

Various techniques and instruments were used for data collection in this field research, including both direct and indirect observation, and tools such as observation guides and reports.

### 3.2.5 Hypothesis Statement

The following hypotheses were proposed:

H1: Developing a Mobile Application increases the number of Qualified Prospects in apartment sales management at NERMAR real estate.

H2: Developing a Mobile Application reduces the Time for Presentation and Demonstration in apartment sales management at NERMAR real estate.

H3: Developing a Mobile Application increases the Number of Closed Sales in apartment sales management at NERMAR real estate.

For hypothesis testing, the following statistical approach was adopted for each indicator:

$\mu_1$ = Population mean (H1, H3) for Post-Test Gc.

$\mu_2$ = Population mean (H1, H3) for Post-Test Ge.

where:

$$Ho: \mu_1 < \mu_2 \quad \text{and} \quad Ha: \mu_1 \geq \mu_2.$$

Additionally:

$\mu_1$ = Population mean (H2) for Post-Test Gc.

$\mu_2$ = Population mean (H2) for Post-Test Ge.

where:

$$H0: \mu_1 \leq \mu_2 \text{ and } Ha: \mu_1 > \mu_2.$$

The data normality test was conducted, followed by descriptive statistical analysis (See Tables 6 and 7), and the hypotheses were validated using the Student's t-test with Minitab software.

### 3.2.6 Solution Development Methodology

This section provides a detailed overview of how the proposed solution was designed and created for the study. It describes the steps, tools, and strategies employed to develop the solution, whether it be a mobile application, software, or any other product.

This section is crucial for understanding how the research problem was approached and solved, ensuring the replicability and effectiveness of the proposed solution.

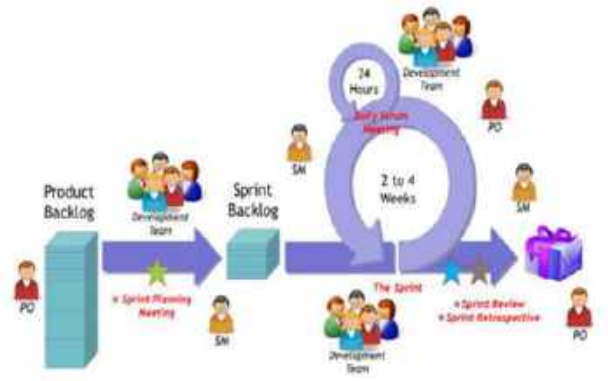



**Fig. 2.** Scrum framework

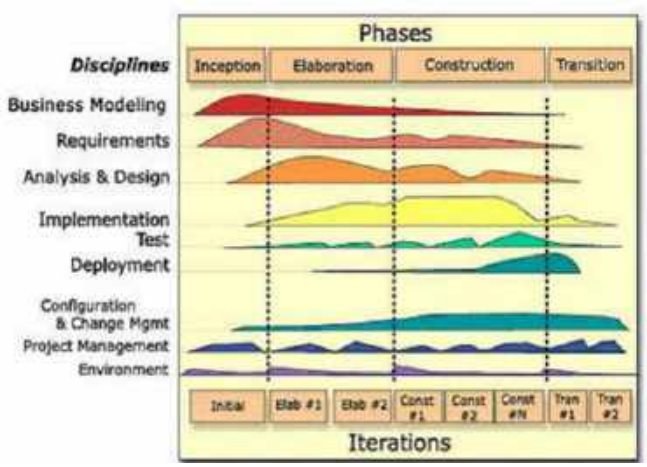



**Fig. 3.** RUP phases

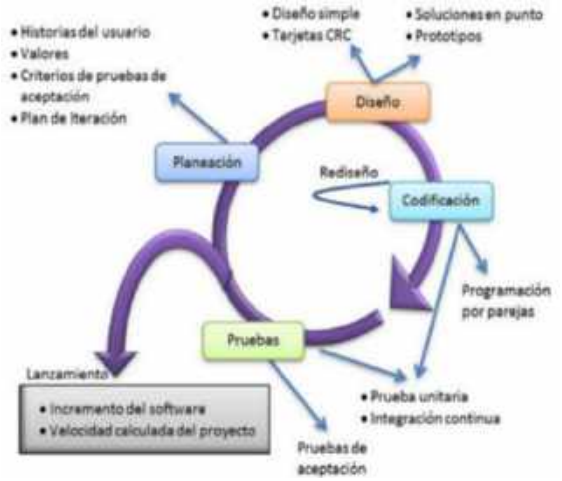



**Fig. 4.** XP phases

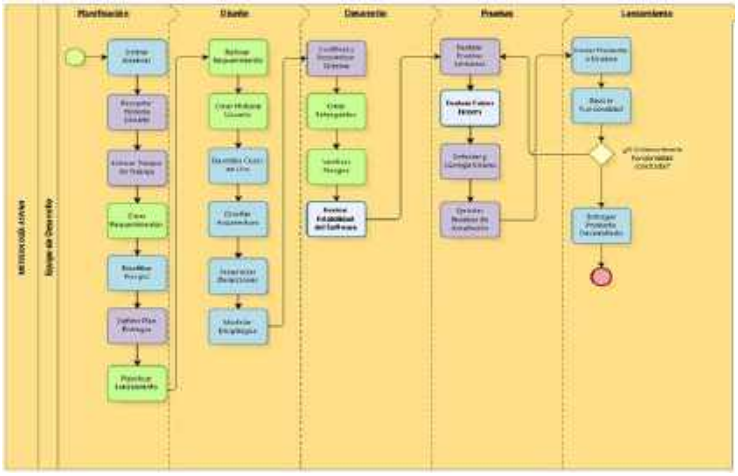



**Fig. 5.** Flowchart of the new ASWAN methodology

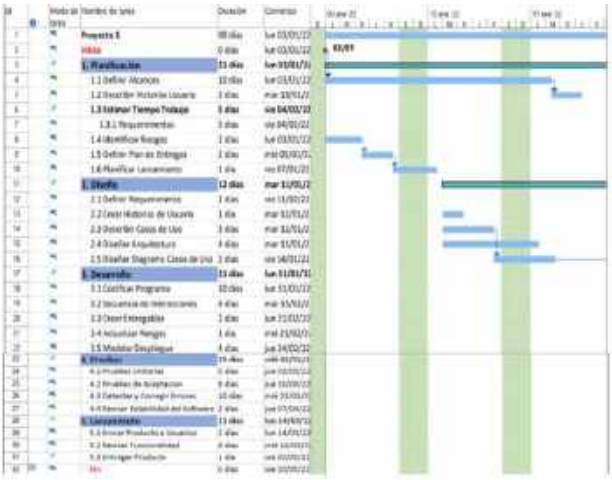


**Fig. 6.** Planning schedule

### 3.2.7 Reviewed Methodologies

The collaborative metamodel ASWAN has been designed incorporating the foundations of the following agile methodologies: Scrum, Rational Unified Process (RUP), and Extreme Programming (XP).

– **Scrum**

Scrum is a framework that aids in generating value through adaptable solutions to complex problems, comprising a Scrum Master, Product Owner, and the development team. (See Figure 2).

– **Rational Unified Process (RUP)**

The Rational Unified Process (RUP) focuses on planning, modeling, implementing, and monitoring the software development lifecycle, providing a robust framework for project management and quality control. (See Figure 3).

– **Extreme Programming (XP)**

The XP methodology defines four variables: cost, time, quality, and scope. It employs short development cycles called iterations, with functional deliverables at the end of each cycle. It consists of four phases, (See Figure 4).

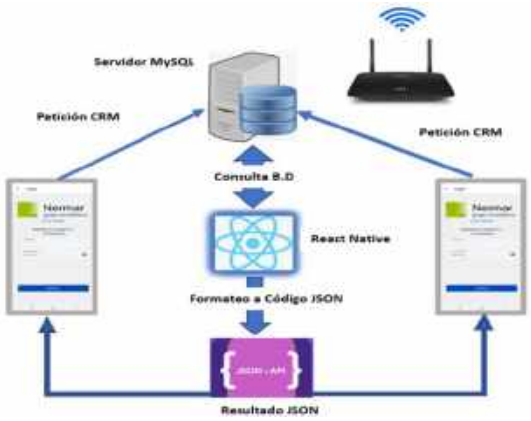



**Fig. 7.** Application architecture


**Fig. 8.** Presentation and login

#### 3.2.8 Collaborative Metamodel: ASWAN

– **Methodology**

Following a thorough analysis of existing models, a new model was created, and the flowchart of the new ASWAN methodology is presented. (See Figure 5).

## 4 Case Study

The Collaborative Metamodel ASWAN, which comprises five phases: Planning, Design, Development, Testing, and Release, was applied.

### 4.1 Phase 1: Planning

The aim of this phase is to estimate technological resources, time, and project risks. It also involves detailing User Stories and preparing the Deliverables Plan. The project planning schedule is displayed (See Figure 6).

### 4.2 Phase 2: Design

This phase focuses on designing the solution's architecture; it involves defining component requirements, software, User Stories, Use Cases, and creating UML Diagrams. Deployment diagram

**Table 2**. Risk management structure

| Risk Management | Risk Analysis | |
|---|---|---|
| **Risks** | **Probability** | **Impact** |
| **Project Risks** | | |
| Unrealistic budgets | Moderate | Serious |
| Inadequate personnel | Moderate | Critical |
| Resource loss | Moderate | Critical |
| Client requirement changes | Moderate | Critical |
| Incomplete requirements | Moderate | Critical |
| **Technical Risks** | | |
| Design, design rework required | Moderate | Serious |
| Implementation, component deficiencies | High | Critical |
| Interface, erroneous development | High | Critical |
| **Business Risks** | | |
| Market, country risk change | High | Critical |
| Strategy, management direction changes | Low | Serious |
| Management, financial issues | Low | Serious |
| Budget, mistaken estimation | Low | Serious |

of the mobile solution and application architecture (See Figure 7).

Figures 8 show the application prototypes on the presentation and login screens.

Figure 9 describes how the application allows managing and organizing schedules to communicate with the buyer and provide sales condition information as quickly as possible. The interface for scheduling and setting times.

### 4.3 Phase 4: Development

This phase involves coding the system and creating its versions. Additionally, the iteration sequences that will guide the development are planned, displaying a version of the implementation diagram (See Figure 10).

– **Risk Management**

During the software development lifecycle, Project Risks, Technical Risks, and Business Risks were considered; the analysis criteria were: Probability: High, Moderate, and Low; impact level: Serious and Critical. The Risk Management structure is shown in Table 2.

### 4.4 Phase 4: Testing

During this stage, Test Plans were developed throughout the system development cycle; both White Box and Black Box testing were used to evaluate functionality, installation, compatibility, usability, and network connection.

All these aspects were aimed at measuring the following attributes: Availability, Robustness, Usability, Configuration, Maintenance, and Security of the mobile application.

– **White Box Testing**

It shows the tests performed on user login, path graph: Login (See Table 3 and Figure 11).


**Fig. 9.** Schedule and set time

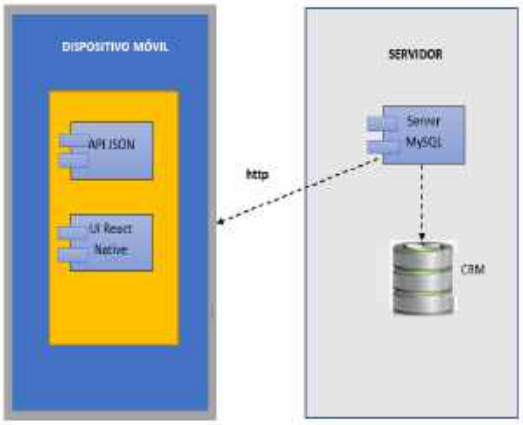



**Fig. 10.** Implementation diagram

– **Black Box Testing**

– This testing assumed the scenarios indicated (See Table 4 and Figure 12).

– **4.5 Phase 5: Release**

– The release phase is a crucial stage in the project development cycle, as it involves implementing and launching the system in a production environment. During this phase, extensive testing is carried out to ensure the stability and functionality of the system, and any emerging issues are addressed.

– Additionally, training is provided for the staff who will use the system, and procedures are established for technical support and ongoing monitoring.

– This section of the article highlights the importance of the outcomes and findings obtained, which become the essential foundation for understanding the transition from development to the effective operation and use of the system.

– **Table 3**. White box testing user login

| White Box Testing: Login Code | | |
|---|---|---|
| | No. of Paths | Independent Path Routes |
| Formula<br>V(G)=P+1=2+1=3<br>V(G)=Cyclomatic Complexity<br>P=Predicate Nodes | 1 | 1 – 5 - 6 |
| | 2 | 1 – 2 – 3 - 6 |
| | 3 | 1 – 5 – 4 – 2 - 3 - 6 |

– **Table 4**. Black box testing scenario

| ID_CODE_PR | SCENE | USER | PASS-WORD | RESULT |
|---|---|---|---|---|
| CÓD_PR_1 | SCENE | | A | Enter password |
| CÓD_PR_2 | SCENE | ADM | | Enter user |
| CÓD_PR_3 | SCENE | | | Enter password |

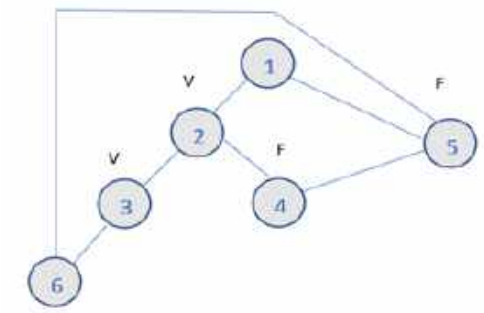
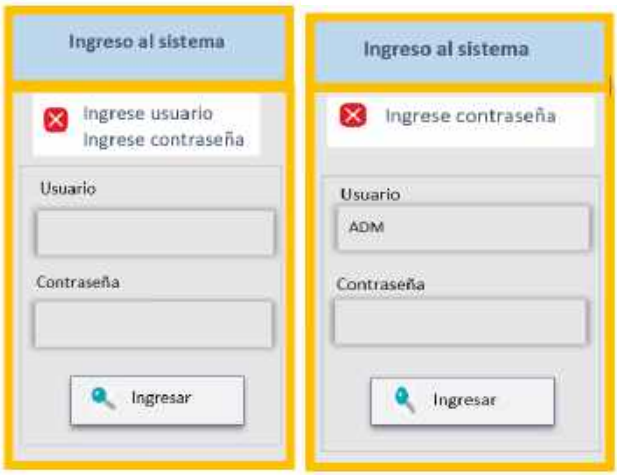



**Fig. 11.** Path graph: Login




– **Fig. 12.** Test case 2 and 3 black box

**Table 5**. Post-test results for Gc and Ge for $I_1$, $I_2$, $I_3$

| **Measurement** | **$I_1$: Number of Qualified Prospects / Leads** | | **$I_2$: Time for Presentation and Demonstration Min.** | | **$I_3$: Number of Closed Sales / Department** | |
|---|---|---|---|---|---|---|
| **Frequency** | **Weekly** | | **Biweekly** | | **Biweekly** | |
| **N°** | **PostTest Gc** | **PostTest Ge** | **PostTest Gc** | **PostTest Ge** | **PostTest Gc** | **PostTest Ge** |
| 1 | 19 | 21 | 19.00 | 14.15 | 8 | 10 |
| 2 | 15 | 20 | 27.00 | 13.00 | 6 | 8 |
| 3 | 10 | 15 | 28.00 | 17.27 | 7 | 9 |
| 4 | 13 | 17 | 24.00 | 14.50 | 7 | 11 |
| 5 | 20 | 22 | 26.00 | 18.20 | 9 | 10 |
| 6 | 16 | 20 | 25.00 | 16.25 | 6 | 8 |
| 7 | 12 | 16 | 22.00 | 15.00 | 10 | 12 |
| 8 | 10 | 14 | 26.00 | 15.00 | 12 | 11 |
| 9 | 14 | 18 | 22.00 | 16.50 | 4 | 6 |
| 10 | 18 | 21 | 25.00 | 17.45 | 7 | 9 |
| 11 | 16 | 20 | 24.00 | 15.30 | 6 | 7 |
| 12 | 15 | 16 | 22.00 | 16.22 | 5 | 6 |
| 13 | 16 | 21 | 21.00 | 14.55 | 8 | 9 |
| 14 | 14 | 19 | 19.00 | 17.42 | 8 | 10 |
| 15 | 10 | 15 | 21.00 | 15.55 | 6 | 8 |
| 16 | 15 | 14 | 29.00 | 14.41 | 5 | 7 |
| 17 | 13 | 16 | 28.00 | 16.54 | 7 | 8 |
| 18 | 14 | 17 | 27.00 | 15.47 | 4 | 6 |
| 19 | 11 | 14 | 23.00 | 17.56 | 8 | 9 |
| 20 | 10 | 17 | 22.00 | 14.55 | 7 | 8 |
| 21 | 12 | 15 | 25.00 | 15.44 | 8 | 9 |
| 22 | 16 | 20 | 24.00 | 16.49 | 6 | 8 |
| 23 | 11 | 15 | 27.00 | 18.54 | 5 | 6 |
| 24 | 14 | 16 | 21.00 | 17.43 | 6 | 8 |
| 25 | 11 | 15 | 26.00 | 15.55 | 10 | 12 |
| 26 | 13 | 18 | 28.00 | 18.44 | 5 | 8 |
| 27 | 12 | 17 | 29.00 | 15.49 | 7 | 10 |
| 28 | 17 | 19 | 27.00 | 17.43 | 5 | 7 |
| 29 | 14 | 19 | 27.00 | 16.30 | 6 | 8 |
| 30 | 13 | 18 | 24.00 | 15.38 | 4 | 10 |

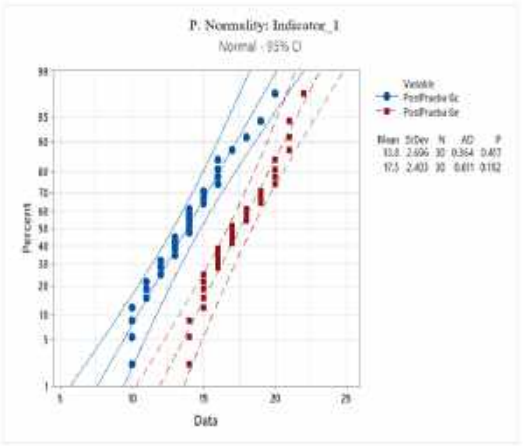



**Fig. 13.** Normality test for indicator I1

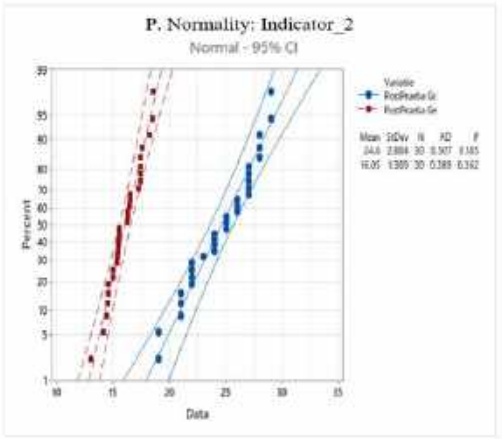



**Fig. 14.** Normality test for indicator I2

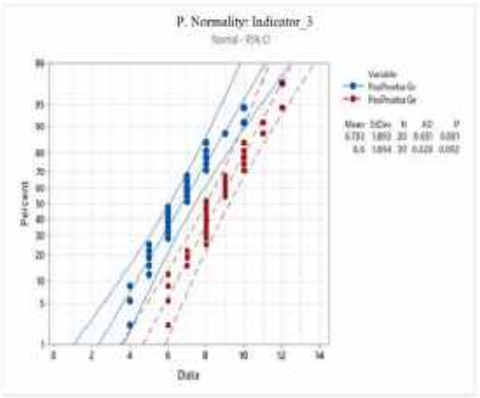



**Fig. 15.** Normality test for indicator I3

– The implementation of a software development metamodel called ASWAN is proposed as a solution to enhance apartment sales management in the real estate context. This approach promises to offer a solid basis for decision-making and effective system management in this specific sector.

## 5 Results and Discussion

Reduction in Time to Contact Prospects, Time of Presentation and Demonstration, Number of Prospects Withdrawn. Increase in Number of Qualified Prospects, Number of Closed Sales.

### 5.1 Experimental Results

For our purposes, measurements were taken from two groups: the control group and the experimental group. Employing statistical techniques, 30 values were obtained for each of the indicators established in the study. The results are detailed in Table 5. Undoubtedly, the implementation of the innovative collaborative metamodel ASWAN yielded positive results in all three indicators.

**Table 6**. Descriptive statistics results of quantitative indicators

| Indicator | n | Mean | StDev | AD | p-value |
|---|---|---|---|---|---|
| I1:PostTest Gc | 30 | 13.80 | 2.696 | 0.364 | 0.417 |
| I1:PostTest Ge | | 17.50 | 2.403 | 0.611 | 0.102 |
| I2:PostTest Gc | 30 | 24.60 | 2.884 | 0.507 | 0.185 |
| I2:PostTest Ge | | 16.05 | 1.389 | 0.389 | 0,362 |
| I3:PostTest Gc | 30 | 6.7 | 1.893 | 0.651 | 0.081 |
| I3: PostTest Ge | | 8.6 | 1.694 | 0.628 | 0.092 |

### 5.2 Normality Test

This section examines the application of the normality test to the data collected in the study. This statistical test is essential for assessing the data distribution and determining if it fits a normal distribution for each indicator. (See Figures 13 to 15).

**For I1:** Number of Qualified Prospects.

**For I2:** Time for Presentation and Demonstration.

**For I3:** Number of Closed Sales.

In the case of indicators (I1, I2, I3), the analysis reveals a p-value that exceeds the predefined significance level (α = 0.05). The data satisfy the assumption of a normal distribution. Given this finding, we have decided to use the parametric Student's t-test to verify the hypothesis.

### 5.3 Discussion of Results

The discussion is carried out with a critical and honest behavior by the researchers, respecting the original results, without altering or distorting them.

#### 2.3.1 With Descriptive Statistics

Provides a detailed analysis of the data collected in the study. It summarizes the main characteristics and trends of the variables studied, allowing for a clear and concise understanding of the information. (See Tables 6 and 7).

Anderson-Darling Normality Test results and the p-value are greater than α (0.05), confirming the normality of the data. Moreover, with a 95% confidence level, the mean and standard deviation values show normal results for each of the research indicators.

The summary of statistical analyses reveals that approximately 95% of the values for each indicator fall within 2 standard deviations of the mean. Kurtosis suggests tails with a lower density than a normal distribution, and the third quartile (Q3) indicates that 75% of the values are equal to or less than this cutoff point.

**For Indicator 1**: Number of Qualified Prospects of the Post-Test Ge group, are within 2 standard deviations of the mean (16.603 and 18.397); kurtosis indicates tails with less density than the normal distribution (-1.17839); the 3rd quartile 20, indicates that 75% of the values are less than or equal to this value, improving the results of the Post-Test Gc group, a similar result was obtained by the authors [16], who managed to increase the number of potential clients, demonstrated by the statistical T Student test, with a significance level of 5% and a confidence level of 95%, obtaining a result of 162 before the implementation of the mobile application and 248 after the implementation, which meant an increase of 86%.

According to the research [21], buyers have obtained more information about the housing construction project with the application, they can ask questions about the project or request interior design services. Therefore, it is more likely that

**Table 7**. Summary of results for quantitative indicators

| Sample | n | 95% Confidence Intervals for the Mean | Kurtosis | Skewness | Q3 |
|---|---|---|---|---|---|
| I1: PostTest Ge | 30 | 16.603 -18.397 | -1.17839 | 0.19163 | 20 |
| I2: PostTest Ge | 30 | 15.527 - 16.565 | -0.59446 | -0.25510 | 17.42 |
| I3: PostTest Ge | 30 | 7.9675 – 9.2325 | -0.46892 | 0.27187 | 10 |

**Table 8**. Summary of hypothesis testing

| Sample | n | $H_0$ | t-value | p-value |
|---|---|---|---|---|
| I1: PostTest(Gc) / I1: PostTest(Ge) | 30 | μ1 ≥ μ2 | -5.61 | 0.000 |
| I2: PostTest(Gc) / I2: PostTest(Ge) | 30 | μ1 ≤ μ2 | 14.64 | 0.000 |
| I3: PostTest(Gc) / I3: PostTest(Ge) | 30 | μ1 ≥ μ2 | -4.03 | 0.000 |

buyers get an ideal home and are happy with their purchase.

The research provides information on the utility of using a mobile application in real estate sales management. This application optimizes internal processes and improves the qualification of prospects. Therefore, the development of a mobile application can be replicated in various management areas, improving job performance, streamlining tasks, and contributing to the improvement of management results.

**For the I2 indicator:** Time for Presentation and Demonstration, it is within two standard deviations of the mean (15.527 and 16.565); kurtosis indicates tails with less density than the normal distribution (-0.59446); the 3rd quartile 17.42, indicates that 75% of the values are less than or equal to this value; times are reduced from the Post-Test Gc group; in the study [21] the average number of daily visits (showroom) was 5.166 during the conventional purchase process and the average number of visits during the innovative purchase process (with the application) was 168,842.

The number of visits increases significantly when customers can interact with a seller through an app instead of traveling to a showroom. The results obtained demonstrate that the mobile solution can be applied in small and medium-sized companies of any economic sector; it will allow managers to make decisions to reduce the times of exposure and presentation of a product and improve their management indicators.

A reference framework is offered, in which a scientifically tested mobile application is developed, with an effect on the organization's results.

**For the I3 indicator:** Number of Closed Sales, the results obtained show that around 95% of the values are within two standard deviations of the

mean (7.9675 and 9.2325); kurtosis indicates tails with less density than the normal distribution (-0.46892); the 3rd quartile 10, indicates that 75% of the values are less than or equal to this value; better results than the Post-Test Gc group.

There is an increase in this indicator with the mobile application, similar results were obtained, research [21] improved the average number of daily sales indicating that it increased from 3.91 to 19.94, which reflects a significant increase. Respondents also have a positive attitude indicating a standard deviation of 1.361 and mean standard of 1.108.

The authors in [16] managed to increase the level of service, demonstrated by the Wilcoxon statistical test with a significance level of 5% and a confidence level of 95% obtaining a result of 95.21% before the implementation of the multiplatform mobile application and 98.66% after implementation, which meant an increase of 3.45.

The findings of the research contribute to the analysis of the information of the sales activities of a real estate company, involved and interested in increasing its sales level and improving its processes through mobile technology. For this purpose, a framework is presented that substantially contributes to the knowledge that well-used digital technology affects the performance and staff and creates value for the company's product and/or service.

### 5.2.1 With Inferential Statistics

Table 8 shows the values from the application of statistical tests performed on each of the indicators for the hypothesis testing.

The results of the parametric tests for the 3 proposed hypotheses indicate a p-value less than α (0.05) in each case, which provides sufficient statistical evidence to reject the null hypotheses Ho, making the alternative hypotheses true. The tests are significant. It is important to mention the scarcity of other case studies in companies of this sector, a situation that has limited the comparison of the Hypothesis Testing data Ho for each indicator, with other companies.

The study's results will provide information for the real estate sector involved in improving their sales management indicators. The results obtained provide enough statistical evidence to reject the Null Hypothesis (H0), and the Alternative Hypothesis (Ha) is true for each indicator. The tests turned out to be significant for each of the proposed indicators.

## 6 Conclusions and Future Research

In the study conducted to improve the process of Sales Management of Apartments in real estate; by integrating with the construct of the Aswan metamodel using foundations from the agile methodologies Scrum, RUP, and XP. The product developed is a technological innovation in the real estate sector; there is no application that measures internal management process indicators.

The results obtained with the Case Study demonstrated that the Aswan Methodology allows the development of a mobile application and improves management indicators; the record of the Presentation and Demonstration Time (I2) using the application was 17.42 minutes, a reduction of 35% compared to the record of 27 minutes without the application, these results had an effect on the number of closed sales (I3) that recorded 10 apartments using the solution, showing an increase of 2 apartments compared to the record of 8 without the solution. Therefore, with these results, it is possible to optimize times and improve sales with the consequent benefits to the involved parties.

The study conducted can be extended in the field of knowledge management, innovation, and software development by adding new AI technologies; the use of the Aswan metamodel is suggested in suitable environments and economic sectors seeking to have a competitive advantage and seek to add value to their products and internal processes.

## References

1. **Hunneman, A., Elhorst, J. P., Bijmolt, T. H. A. (2022).** Store sales evaluation and prediction using spatial panel data models of sales components. Spatial Economic Analysis, Vol. 17, No. 1, pp. 127–150. DOI: 10.1080/17421772.2021.1916574.
2. **Fu, B. (2022).** Does PropTech facilitate liquidity in the property transaction process?

A qualitative study on the swedish real estate market. KTH Royal Institute of Technology.

3. **Pesantez-Nieto, B., Sánchez-Cumbicos, G., Villavicencio-Rodas, M. (2021).** Comportamiento del consumidor en pandemia, una mirada desde el sector inmobiliario de la provincia de El Oro. 593 Digital Publisher CEIT, Vol. 6, No. 6, pp. 293–305. DOI: 10.33386/593dp.2021.6.765.
4. **Meyer, C., Cuchillac, V. (2019).** Uso de tecnologías de programación visual para el desarrollo de aplicaciones móviles. Casos Universidad Francisco Gavidia-UFG. Realidad y Reflexión, Vol. 50, No. 50, pp. 82–107. DOI: 10.5377/ryr.v50i50.9096.
5. **Ullah, F., Sepasgozar, S., Thaheem, M. J., Al-Turjman, F. (2021).** Barriers to the digitalisation and innovation of australian smart real estate: A managerial perspective on the technology non-adoption. Environmental Technology & Innovation, Vol. 22, p. 101527. DOI: 10.1016/j.eti.2021.101527.
6. **Ullah, F., Sepasgozar, S., Jamaluddin-Thaheem, M., Cynthia-Wang, C., Imran, M. (2021).** It's all about perceptions: A DEMATEL approach to exploring user perceptions of real estate online platforms. Ain Shams Engineering Journal, Vol. 12, No. 4, pp. 4297–4317. DOI: 10.1016/j.asej.2021.04.023.
7. **Ullah, F., Sepasgozar, S., Shirowzhan, S., Davis, S. (2021).** Modelling users' perception of the online real estate platforms in a digitally disruptive environment: An integrated KANO-SISQual approach. Telematics and Informatics, Vol. 63, p. 101660. DOI: 10.1016/j.tele.2021.101660.
8. **Makkonen, H., Komulainen, H. (2018).** Explicating the market dimension in the study of digital innovation: a management framework for digital innovation. Technology Analysis & Strategic Management, Vol. 30, No. 9, pp. 1015–1028. DOI: 10.1080/09537325.2018.1433823.
9. **Seya, H., Shiroi, D. (2022).** A comparison of residential apartment rent price predictions using a large data set: kriging versus deep neural network. Geographical Analysis, Vol. 54, No. 2, pp. 239–260. DOI: 10.1111/gean.12283.
10. **So, H., Chung, W. (2005).** Mobile IT infrastructure in value network development: A case study of property management business. Production Planning & Control, Vol. 16, No. 6, pp. 586–596. DOI: 10.1080/09537280500112116.
11. **Lai, J., Wang, J., Ulhas, K., Chang, C. (2022).** Aligning strategy with knowledge management system for improving innovation and business performance. Technology Analysis & Strategic Management, Vol. 34, No. 4, pp. 474–487. DOI: 10.1080/09537325.2021.1907328.
12. **Al-Maamzi, J., Tawfik, T. (2022).** The effectiveness of agile management on traditional projects within public organizations. IOP Conference Series: Materials Science and Engineering, Vol. 1218, No. 1, p. 012037. DOI: 10.1088/1757-899x/1218/1/012037.
13. **Choto Maza, J., Avila, D., Avila Pesantez, L. (2020).** Desarrollo de una aplicación móvil utilizando el framework MEAN Stack e IONIC: Un estudio de caso en una compañía de transporte. Ecuadorian Science Journal, Vol. 4, No. 2, pp. 37–42. DOI: 10.46480/esj.4.2.74.
14. **Molina-Ríos, J., Honores-Tapia, J., Pedreira-Souto, N., Pardo-León, H. (2021).** Estado del arte: metodologías de desarrollo de aplicaciones móviles. 3C Tecnología. Innovación Aplicada a la Pyme, Vol. 10, No. 2, pp. 17–45. DOI: 10.17993/3ctecno/2021.v10n2e38.17-45.
15. **Molina Ríos, J., Honores Tapia, J., Pedreira-Souto, N., Pardo León, H. (2021).** Comparativa de metodologías de desarrollo de aplicaciones móviles. 3C Tecnología. Innovación Aplicada a la Pyme, Vol. 10, No. 2, pp. 73–93. DOI: 10.17993/3ctecno/2021.v10n2e38.73-93.
16. **Reyna, J., Cieza, S., Alcántara, O. R., Pacheco, J. F. (2021).** Multiplatform mobile application to improve sales management in 'Janavet' vet from Trujillo. Proceedings of the LACCEI International Multi-conference for Engineering, Education, and Technology, Scopus, pp. 1–11.

17. **Hsu, K., Chao, J. (2022).** The impact of urban green-infrastructure development on the price of surrounding real estate: A case study of taichung city's central district. IOP Conference Series: Earth and Environmental Science, Vol. 1006, No. 1. DOI: 10.1088/1755-1315/1006/1/012012.
18. **Ma, L., Liu, H., Edwards, D., Sing, M. (2021).** Housing price dynamics on residential construction: A case study of the Australian property sector. Structural Change and Economic Dynamics, Vol. 59, pp. 525–532. DOI: 10.1016/j.strueco.2021.10.001.
19. **Franco, M., Godinho, L., Rodrigues, M. (2021).** Exploring the influence of digital entrepreneurship on SME digitalization and management. Small Enterprise Research, Vol. 28, No. 3, pp. 269–292. DOI: 10.1080/13215906.2021.1938651.
20. **Díaz-Martínez, M., Ruíz-Hernández, S., Román-Salinas, R., Estrada-Cadena, G. (2021).** Aplicación móvil 'AppIndustria 4.0': una herramienta para la evaluación de las organizaciones en industria 4.0. Información Tecnológica, Vol. 32, No. 4, pp. 53–64. DOI: 10.4067/S0718-07642021000400053.
21. **Levi-Bliech, M., Pliskin, N., Fink, L. (2020).** Implementing a sales support app to complement face-to-face interaction: An empirical investigation of business value. Journal of Organizational Computing and Electronic Commerce, Vol. 30, No. 3, pp. 266–278. DOI: 10.1080/10919392.2020.1750932.
22. **De-Clase, N., Vélez, A. (2007).** El proceso de desarrollo de software. Campus Virtual Univalle, Vol. 3, No. 1, pp. 131–146.
23. **Galindo, N., De-Moura, G., Barbosa, L., De-Castro, J., Barros, A., Moreira, L. (2020).** COVID-19 and digital technology: Mobile applications available for download in smartphones. Texto e Contexto Enfermagem, Vol. 29, pp. 1–11. DOI: 10.1590/1980-265x-tce-2020-0150.
24. **Lavrov, N., Druzhinin, A., Alekseeva, N. (2020).** Description and analysis of economic efficiency of the real estate model transformed in the framework of digitalization. IOP Conference Series: Materials Science and Engineering, Vol. 940, No. 1, pp. 1–11. DOI: 10.1088/1757-899X/940/1/012042.
25. **Siniak, N., Kauko, T., Shavrov, S., Marina, N. (2020).** The impact of proptech on real estate industry growth. IOP Conference Series: Materials Science and Engineering, Vol. 869, No. 6, pp. 1–12. DOI: 10.1088/1757-899X/869/6/062041.
26. **Kumar, R. (2021).** Managing business in the digital era–the use of it and analytics for process transformation. Journal of Decision Systems, Vol. 30, No. 4, pp. 410–413. DOI: 10.1080/12460125.2021.1925397.
27. **Thielen, T., Decramer, A., Vanderstraeten, A., Audenaert, M. (2022).** The effects of performance management on relational coordination in policing: the roles of content and process. International Journal of Human Resource Management, Vol. 33, No. 7, pp. 1377–1402. DOI: 10.1080/09585192.2020.1779779.
28. **Daul, V., Illescas, W., Herrera, E., Lara, P. (2019).** Análisis de tendencias inmobiliarias de los mercados en vías de desarrollo desde la perspectiva tecnológica. Centro de Investigaciones UTMCH, 2do Congreso Internacional Tecnologías para el Desarrollo, Vol. 2, pp. 9–25.
29. **Chung, W., Mustaine, E., Zeng, D. (2021).** A computational framework for social-media-based business analytics and knowledge creation: empirical studies of CyTraSS. Enterprise Information Systems, Vol. 15, No. 10, pp. 1460–1482. DOI: 10.1080/17517575.2020.1827299.
30. **Fan, Y., Leung, C., Yang, Z. (2022).** Financial conditions, local competition, and local market leaders: The case of real estate developers. Pacific Economic Review, Vol. 27, No. 2, pp. 131–193. DOI: 10.1111/1468-0106.12360.
31. **Liu, Y., Wang, P., He, Z., Dong, J. (2021).** Real estate rental market: a 10-year bibliometric-based review. Economic Research-Ekonomska Istraživanja, Vol. 34, No. 1, pp. 1752–1788. DOI: 10.1080/1331677X.2020.1848605.
32. **Peng, Z., Yu, Z., Nong, H. (2020).** Inter-type investment connectedness: A new

perspective on china's booming real estate market. Global Economic Review, Vol. 49, No. 2, pp. 186–204. DOI: 10.1080/1226508X.2020.1744465.

33. **Chiclayo-Silvestre, A., Nizama-Florian, J., Gamboa-Cruzado, J., Vásquez-Valencia, Y., Tello-Aguilar, C. (2024).** Maskay-aplicación móvil para la búsqueda y acceso a talleres mecánicos: Un estudio de caso en Trujillo, Perú. RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação, No. E65, pp. 204–209.

34. **Angulo-Pizan, E., Tello-León, J., Gamboa-Cruzado, J., Vásquez-Valencia, Y., Salas-Ruiz, J. (2024).** Yuyachiy-aplicación móvil para la atención de emergencias: Un estudio de caso en la municipalidad distrital de Chicama, Perú. RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação, No. E65, 1–16.

*Article received on 04/06/2024; accepted on 12/09/2024.*
**Corresponding author is César Jesús Núñez-Prado.*

ISSN 2007-9737

# Social Influence, Market Manipulators, Hardware and Software as New Factors for Cryptocurrency Pricing: A Survey

Bulat Shkanov[1], Mikhail Alexandrov[2,3,*]

[1] Gaidar Institute for Economic Policy, Moscow, Russia

[2] Russian Presidential Academy of National Economy and Public Administration, Moscow, Russia

[3] FRUCT Association Helsinki, Finland

bulat.shkanov@mail.ru, malexandrov@mail.ru

**Abstract.** In the last decade, cryptocurrencies have been confidently setting the trend in global financial markets. Nevertheless, a certain degree of mistrust toward the use of cryptocurrencies persists among investors and researchers due to the lack of reliable models for predicting their dynamics. Such models can be constructed based on a comprehensive consideration of the factors that determine the price of cryptocurrencies. This article provides: (a) a brief overview of known factors affecting cryptocurrency pricing, and (b) a more detailed review of new factors, including those related to computational resources and software tools. This review may be useful for researchers and practitioners building mathematical models for cryptocurrency forecasting.

**Keywords.** Cryptocurrency pricing, social factors, market manipulators, hardware-technological factors, software-algorithmic factors.

## 1 Introduction

In recent years, cryptocurrencies have become an integral part of the global financial system, attracting the attention of both investors and regulators. Since the emergence of the first cryptocurrency, Bitcoin, in 2009, the market for these digital assets has shown incredible volatility and rapid development. The market capitalization of cryptocurrencies has reached hundreds of billions of dollars in a short period, indicating their significant impact on the economy and global financial infrastructure. However, despite their growing popularity, the mechanisms of cryptocurrency pricing remain complex and insufficiently studied.

Traditional approaches to pricing analysis, which are applied to fiat currencies and classic financial assets, do not always effectively explain the price fluctuations in the cryptocurrency market. This is due to the unique characteristics of crypto assets, such as decentralization, limited supply, transaction anonymity, and a high degree of speculative activity.

As a result of these features, it is necessary to consider not only traditional macroeconomic and financial factors but also new phenomena, such as socio-economic influences, market manipulation, and technological changes.

This article provides an overview of the factors influencing cryptocurrency pricing, emphasizing the need to expand the existing classification of factors to include new, less-studied aspects. Separate sections are devoted to the consideration of hardware-technological and software-algorithmic factors.

Discussing these factors will deepen the understanding of the processes underlying

cryptocurrency price dynamics, which is important for both researchers and market participants.

## 2 Traditional Factors

Traditional factors typically include fundamental, macroeconomic, financial, behavioral, and infrastructural aspects, as identified by the scientific community [1-5].

### 2.1 Fundamental Factors

Fundamental factors reflect basic decisions, such as the adoption of cryptocurrencies as a means of payment and regulatory changes. These factors lay the foundation for cryptocurrency price growth. Short-term fluctuations can be caused by speculation and changes in market sentiment [6].

### 2.2 Macroeconomic Factors

These involve market conditions and macroeconomic indicators, such as liquidity, trading volume, and the influence of social media, which significantly impact the stability and volatility of cryptocurrencies. Regulatory changes also have a strong effect on the market [7].

### 2.3 Financial Factors

This category includes factors affecting the pricing of cryptocurrency options and other financial derivatives, as well as the role of market expectations in cryptocurrency pricing. Additionally, cryptocurrencies can serve as a hedging tool against political and economic uncertainty [8].

### 2.4 Behavioral Factors

Psychological aspects, such as speculation and media attention, contribute to the formation of "bubbles" in the cryptocurrency market. Public sentiment, expressed on social media, can also predict price changes [9].

### 2.5 Infrastructural Factors

Parameters like mining difficulty, hash rate, and the concentration of mining pools also play a significant role in cryptocurrency pricing. Changes in these indicators can affect the scarcity of cryptocurrencies and their volatility [10].

## 3 Social Influence Factors

Free access to the Internet has revealed various manifestations of the influence of socio-economic factors on cryptocurrency pricing. In this review, we highlight two of them: trust in financial institutions and the role of social media.

### 3.1 Level of Trust in Financial Institutions

Social factors, such as the level of trust in financial institutions and the degree of economic freedom, significantly impact the price formation process of cryptocurrencies [11].

Trust in financial organizations plays a crucial role in determining the willingness of individuals and companies to use cryptocurrencies as alternative financial instruments. A higher level of trust leads to increased demand for cryptocurrencies, which in turn positively affects their market value.

The degree of economic freedom is also an important factor influencing the adoption of cryptocurrencies. In countries with a high level of economic freedom, where there are fewer regulatory barriers to trade and investment, cryptocurrencies are more rapidly integrated into the economic structure, serving as means of exchange and capital accumulation. In such conditions, cryptocurrencies become a key element of the financial system, helping to maintain their value at a higher level.

Additionally, in study [8], cryptocurrencies are examined in terms of their applicability and functionality in society. Beyond traditional financial factors, such as volatility, liquidity, and risks, their economic value is also influenced by utility, purpose, and level of adoption in the economy. The more extensively cryptocurrencies are used in real economic operations, the higher their market value.

For example, the use of cryptocurrencies in international remittances, e-commerce, and decentralized finance (DeFi) applications enhances their economic significance and, consequently, increases their market value.

### 3.2 Social Media

Social media plays a vital role in the adoption and dissemination of cryptocurrencies, acting as both an indicator of public sentiment and a factor influencing price formation. The popularity and discussions of cryptocurrencies on platforms like Twitter and Reddit [12, 13, 22] create significant informational impact on market expectations and investor behavior.

Studies show that mentions of cryptocurrencies on social media correlate with price fluctuations, making these platforms important indicators of public acceptance. Discussions of cryptocurrencies on social media not only reflect current societal sentiment but can also provoke large-scale market changes, creating a self-fulfilling prophecy effect. Positive mentions and discussions often lead to increased interest and investments, while negative sentiment can result in decreased trust and falling prices [14, 15].

Moreover, social media facilitates the decentralized spread of information about new projects, cryptocurrencies, and blockchain technologies, making the adoption process more dynamic. Such discussions accelerate the acceptance of cryptocurrencies by influencing public awareness and expanding the user base. Social media also serves as a platform for informational campaigns, educational materials, and discussions, contributing to increased trust in cryptocurrencies and their integration into everyday economic processes.

## 4 Market Manipulators

In a review article by Felix Eigelshoven, André Ullrich, and Douglas Parry [16], various methods of manipulation in the cryptocurrency market and their impact on prices are analyzed. The authors highlight several key methods of market manipulation:

### 4.1 Pump-and-Dump Schemes

This method involves artificially inflating cryptocurrency prices through coordinated buying, followed by a quick sell-off at inflated prices. Manipulators often organize groups on social media to attract inexperienced investors, leading to short-term price spikes. Pump-and-dump schemes contribute to increased volatility and sharp price fluctuations, especially in markets with low capitalization.

### 4.2 Wash Trading

This approach involves creating fake trading volumes through self-dealing, which gives the false impression of high liquidity. It attracts other traders, misleading them about the real trading volumes and market activity.

### 4.3 Insider Trading

The use of non-public information, such as before the official announcement of a new cryptocurrency listing on an exchange, allows traders to gain a competitive advantage. Insider trading undermines market transparency and fairness, potentially leading to sharp price changes in the short term.

### 4.4 DDoS Attacks

These attacks target the temporary shutdown or performance degradation of cryptocurrency exchanges, which can be used to manipulate prices by creating an artificial liquidity shortage. As a result, significant price fluctuations in cryptocurrencies can occur during such attacks.

### 4.5 Manipulations Using Stablecoins

An example of this is the stablecoin Tether, which is issued in large volumes to buy cryptocurrencies during periods of price decline. This temporarily stabilizes the market and supports cryptocurrency prices. However, such manipulations may pose risks to the long-term sustainability of the market.

Market manipulation in the cryptocurrency space remains a significant challenge, negatively affecting investor trust and market stability as a

whole. The main issues stem from a lack of adequate regulation, the anonymity of participants, and the absence of established standards. Addressing these problems requires stricter regulatory measures and increased transparency on cryptocurrency exchanges.

## 5 Hardware-technological Factors

Hardware-technological factors are typically associated only with equipment costs. However, current realities require consideration of other equally significant circumstances. In this review, we identify two subgroups: equipment and mining costs, and halving.

### 5.1 Equipment and Mining Costs

Here, equipment refers to the hardware used for cryptocurrency mining, such as Graphics Processing Units (GPUs), Application-Specific Integrated Circuits (ASICs), and server capacities that support blockchain operations. The efficiency of mining equipment directly impacts parameters like hash rate, which in turn influences the value of cryptocurrencies.

As mining difficulty increases, more powerful equipment is required, leading to higher electricity and equipment costs. This can result in an increase in cryptocurrency prices. The concentration of mining pools and the high energy costs of mining can significantly affect the value of cryptocurrencies.

### 5.2 Halving

Halving is a process embedded in the protocol of most Proof of Work (PoW)-based cryptocurrencies, in which the reward for mining new blocks is halved at regular intervals. This mechanism is used to control cryptocurrency inflation and gradually reduce the number of new coins entering the market.

Halving has a significant impact on cryptocurrency pricing, especially for assets like Bitcoin. As the supply of new coins decreases, if demand remains stable or increases, the cryptocurrency's price can rise substantially. Historically, halvings have been accompanied by significant price fluctuations, including strong increases. This phenomenon can be explained by the economic principle that a restriction in supply amid stable demand leads to an increase in asset value.

A study on Bitcoin [17] showed that the halvings in 2012, 2016, and 2020 were followed by a significant rise in cryptocurrency prices in the subsequent months and years. These events create a supply shortage, stimulating the market and potentially leading to more bullish sentiment among investors.

Halving also changes the economic incentives for miners. With the reduction in block rewards, miners face increased mining costs, which can result in a decrease in the number of participants in mining, especially if the cryptocurrency price does not compensate for the reduced reward. This can also lead to a temporary decrease in hash rate and a slowdown in transaction processing until the balance between supply and demand is restored.

Research indicates that halving creates complex macroeconomic conditions for cryptocurrency markets. In the study by Dyhrberg and other researchers [17], it was noted that halving can lead to long-term price growth due to the creation of a supply shortage, but it can also cause short-term high volatility and uncertainty in the market.

Halving is a key event in the cryptocurrency economy, affecting both the value of coins and the incentives for miners. The impact of halving on prices has been confirmed by historical data and research, and this mechanism continues to be an important factor in analyzing future trends in the cryptocurrency market.

## 6 Software-Algorithmic Factors

Software-algorithmic factors influencing cryptocurrency markets include algorithm improvements, the creation of new blockchains, and the launch of new networks. These technological changes significantly impact the efficiency, security, and popularity of cryptocurrencies, as well as their market value.

### 6.1 Consensus Algorithms

Consensus algorithms such as Proof of Work (PoW) and Proof of Stake (PoS) [18] play a key role in maintaining the security and decentralization of blockchain networks. Improvements to these algorithms or a shift to more efficient protocols can greatly affect network performance, transaction speed, and energy consumption.

For example, Ethereum's transition from PoW to PoS in the Ethereum 2.0 upgrade aimed to reduce energy consumption and improve network scalability [19]. This upgrade not only reduced mining costs but also increased interest in the network by decreasing its environmental footprint and enhancing transaction processing speeds.

Such changes can influence cryptocurrency prices, as improvements in performance and reduced transaction costs make the network more attractive to users and investors. The introduction of new algorithms can also promote the scalability of blockchains, which is crucial for projects aiming for widespread adoption.

### 6.2 New Blockchains

The launch of new blockchains is another important software factor that can affect the cryptocurrency market. For example, the emergence of new networks such as Polkadot and Solana [20], with improved transaction speeds and cross-chain compatibility, has been a significant event for the crypto industry.

These networks offer users and developers enhanced capabilities compared to earlier blockchains like Bitcoin and Ethereum, attracting new investors and developers of decentralized applications (dApps).

Each new network provides unique technical solutions and opportunities for scalability and security. In the case of Polkadot, its parachain system allows separate blockchains to interact with each other, paving the way for complex decentralized ecosystems. This attracts investor attention and increases the value of associated tokens.

### 6.3 New Networks

The emergence of new networks and Layer 2 solutions, such as the Lightning Network [21] for Bitcoin and Optimism for Ethereum, plays an essential role in improving scalability and reducing transaction fees. These solutions operate on top of the main blockchain network, allowing more transactions to be processed without overloading the main blockchain. This reduces costs and enhances the user experience, positively affecting cryptocurrency adoption.

The success of such networks depends on their ability to solve current scalability and transaction speed issues. For instance, the Lightning Network [22] increases the processing speed of microtransactions on the Bitcoin network, making it more appealing for everyday use. Optimism and other Layer 2 solutions help Ethereum handle high fees and slow transaction speeds, increasing its competitiveness among other blockchains.

Software factors, such as consensus algorithm improvements, the creation of new blockchains, and the launch of new networks, are key drivers of the cryptocurrency market's development. These technological innovations contribute to better scalability, reduced costs, and enhanced network security, ultimately influencing the attractiveness of cryptocurrencies for users and investors. The evolution of software factors continues to impact market dynamics and the value of crypto assets.

## 7 Conclusions

The review of cryptocurrency pricing factors covers both traditional and new, relevant aspects influencing their value dynamics. Traditional factors, such as fundamental, macroeconomic, financial, behavioral, and infrastructural elements, continue to play a vital role in forming cryptocurrency prices. However, as cryptocurrency technology and markets evolve, it becomes evident that the existing classification needs to be supplemented with new factors.

Current realities demand consideration of new phenomena, such as socio-economic influences, market manipulation, as well as hardware-technological and software-algorithmic factors,

including blockchain innovations, consensus algorithm improvements, and the emergence of new networks. These new factors significantly impact the market, creating both opportunities and risks for its further development.

Therefore, to gain a deeper understanding of the mechanisms of cryptocurrency pricing and develop reliable forecasting models, it is necessary to update and expand the existing classification of factors. Greater attention should be given to new technological and market trends, which will not only improve the understanding of price dynamics but also enable the development of more accurate risk management strategies in the rapidly growing cryptocurrency market.

## References

1. **Mikhailov, A. Y. (2017).** A theory of cryptoasset valuation. Financial analytics: Science and Experience, Vol. 10, No. 6, pp. 691–700. DOI: 10.24891/fa.10.6.691.
2. **Briere, M., Oosterlinck, K., Szafarz, A. (2017).** Virtual currency, tangible return: portfolio diversification with Bitcoins. CEB Working Papers, Vol. 13, No. 31. DOI: 10.2139/ssrn.2324780.
3. **Popper, N. (2015).** Digital gold: the untold story of Bitcoin. London: Penguin, 432.
4. **Mikhailov, A. Y. (2018).** Pricing on cryptoassets market and influence on fund indices. Financial analytics: problems and solutions, Vol. 24, No. 3, pp. 641–651. DOI: 10.24891/fc.24.3.641.
5. **Kukacka, J., Kristoufek, L. (2020).** Fundamental and speculative components of the cryptocurrency pricing dynamics. Journal of International Financial Markets, Institutions and Money, Vol. 65. DOI: 10.1186/s40854-023-00465-7.
6. **Peng, S., Prentice, C., Shams, S., Sarker, T. (2023).** A systematic literature review on the determinants of cryptocurrency pricing. China Accounting and Finance Review, Vol. 26, No. 1. DOI: 10.1108/CAFR-05-2023-0053.
7. **Corbet, S., Lucey, B., Urquhart, A., Yarovaya, L. (2019).** Cryptocurrencies as a financial asset: A systematic analysis. International Review of Financial Analysis, Vol. 62. DOI: 10.1016/j.irfa.2018.09.003.
8. **Zhang, W., Shen, D., Zhang, Y., Xiong, X. (2013).** Open source information, investor attention, and asset pricing. Economic Modelling, Vol. 33, pp. 613–619. DOI: 10.1016/j.econmod.2013.03.018.
9. **Datta, B., Hodor, I. (2024).** Cryptocurrency, mining pools' concentration, and asset prices. SSRN. DOI: 10.2139/ssrn.3887256.
10. **Shambhavi, S., Kanika. M. (2022).** Social and economic factors that impact cryptocurrency. Indian Journal of Applied Research, pp. 63–68. DOI: 10.36106/ijar/1814049.
11. **Kraaijeveld, O., De-Smedt, J. (2020).** The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. Journal of International Financial Markets, Institutions and Money, Vol. 65, p.101188. DOI: 10.1016/j.intfin.2020.101188.
12. **Coulter, K. A. (2022).** The impact of news media on bitcoin prices: modelling data driven discourses in the crypto-economy with natural language processing. Royal Society Open Science, Vol. 9, No. 4, p. 220276**.** DOI: 10.1098/rsos.220276.
13. **Polasik, M., Piotrowska, A. I., Wisniewski, T. P., Kotkowski, R., Lightfoot, G. (2015).** Price fluctuations and the use of bitcoin: An empirical inquiry. International Journal of Electronic Commerce, Vol. 20, No. 1, pp. 9–49**.** DOI: 10.1080/10864415.2016.1061413.
14. **Shahzad, M. F., Xu, S., Lim, W. M., Hasnain, M. F., Nusrat, S. (2024).** Cryptocurrency awareness, acceptance, and adoption: the role of trust as a cornerstone. Humanities and Social Sciences Communications, Vol. 11, No. 1, pp. 1–14. DOI: 10.1057/s41599-023-02528- 7.
15. **Eigelshoven, F., Ullrich, A., Parry, D. (2021).** Cryptocurrency market manipulation-A systematic literature review. ICIS.
16. **Dyhrberg, A. H. (2016).** Hedging capabilities of bitcoin. Is it the virtual gold? Finance Research Letters, Vol. 16, p. 139–144. DOI: 10.1016/j.frl.2015.10.025.

17. **Auer, R. (2019).** Beyond the doomsday economics of 'proof-of-work' in cryptocurrencies. BIS Working Paper No. 765.
18. **Buterin, V. (2020).** Ethereum 2.0 and the future of Blockchain. Ethereum Foundation Blog. URL:https://getintopc.com/reviews/exploring-the-ethereum-2-0-upgrade-impact-on-eth-price-and-the-future-of-cryptocurrency/.
19. **Wood, G. (2016).** Polkadot: Vision for a heterogeneous multi-chain framework. White paper, Vol. 21, No. 2327, p. 4662.
20. **Sguanci, C., Spatafora, R., Vergani, A. M. (2021).** Layer 2 blockchain scaling: A survey. arXiv preprint arXiv:2107.10881. DOI: 10.48550/arXiv.2107.108.
21. **Divakaruni, A., Zimmerman, P. (2022).** The lightning network: turning bitcoin into money. Federal Reserve Bank of Cleveland, Working Papers, Vol. 22-19, pp. 1–18. DOI: 10.26509/frbc-wp-202219.
22. **Tash, M.S., Kolesnikova, O., Ahani, Z., Sidorov, G. (2024).** Psycholinguistic and emotion analysis of cryptocurrency discourse on X platform. Sci Rep 14, 8585. DOI: 10.1038/s41598-024-58929-4.

*Article received on 03/05/2024; accepted on 07/07/2024.*
**Corresponding author is Mikhail Alexandrov.*

ISSN 2007-9737

# Semantic Textual Similarity: Overview and Comparative Study between Arabic and English

Samira Boudaa*, Tarik Boudaa, Anass El Haddadi

Abdelmalek Essaadi University,
National School of Applied Sciences,
LSA Laboratory, Al Hoceima,
Morocco

samira.boudaa@etu.uae.ac.ma, t.boudaa@uae.ac.ma, a.elhaddadi@uae.ac.ma

**Abstract.** Semantic Textual Similarity is crucial for various end-user applications of Natural Language Processing, including Search Engines, Chatbots, Machine Translation Systems, Plagiarism Detection, and Text Summarization. While substantial research has been conducted on this topic for widely spoken languages such as English, there exists a need for comprehensive surveys focusing on less-studied languages, such as Arabic. This work is a comprehensive resource for researchers working on Semantic Textual Similarity especially for the Arabic language. Our survey synthesizes the current state of research in Semantic Textual Similarity in Arabic, providing valuable insights into this field's unique challenges and opportunities. We review state-of-the-art approaches, datasets, and methodologies proposed for Arabic Semantic Textual Similarity. The paper highlights the differences between Arabic and English, which necessitate tailored approaches to Semantic Textual Similarity. Moreover, we discuss the recent advancements in Arabic Semantic Textual Similarity and identify the existing gaps and challenges that researchers face. In addition, we propose potential future research directions to further improve the Arabic Semantic Textual Similarity models. By addressing these areas, our work aims to foster a deeper understanding and more robust development of Semantic Textual Similarity for the Arabic language, ultimately expanding the scope and effectiveness of Semantic Textual Similarity applications.

**Keywords.** Semantic textual similarity, question similarity, Arabic NLP.

## 1 Introduction

In NLP, Semantic Textual Similarity (STS) is a task that aims to quantify the degree of similarity between two texts at the semantic level rather than just their surface form. This similarity taking into account the semantics and contextual meaning is essential for tasks requiring deeper understanding of textual data, such as Information Retrieval (e.g., [1]), Machine Translation (e.g., [2]), Text Summarization (e.g., [3]), Question Answering (e.g., [4]), and Plagiarism Detection (e.g., [5, 6]).

In essence, STS encompasses a hierarchy of tasks, ranging from word-level comparisons to sentence-level analysis and document-level assessments, each contributing to achieve a deeper comprehension of text similarity and enabling various applications across the field of NLP.

STS has a crucial role in the evaluation of language models, and word and sentence representation models. The performance of these models is judged through their performance in STS Benchmark.

While the methods for STS have seen important progress for English, there exists a convincing need to explore the state of the art for languages with distinct linguistic characteristics and cultural backgrounds, such as Arabic. A survey on Arabic STS remains highly relevant despite some existing survey works [7, 8, 9].

Indeed, while these previous works provided valuable insights into the state of STS research for the Arabic language, they miss the latest developments and breakthroughs that have emerged in NLP through deep learning techniques. The present survey attempts to highlight recent works, advancements, challenges, and opportunities, enabling a more comprehensive

understanding of the current landscape and inspiring further innovation in this important area of NLP.

It explores the various techniques and methodologies employed to tackle this task, ranging from traditional lexical and distributional approaches to modern deep learning-based methods. By doing so, we aim to shed light on the strengths and weaknesses of each approach in the specific context of the Arabic language. We also review the available Arabic STS datasets.

Furthermore, we address the unique challenges associated with Arabic STS, including the complexity of Arabic morphology, the presence of dialectal variations, and the scarcity of annotated data. We also explore recent advancements and innovations that have emerged to address these challenges, as well as propose potential avenues for future research exploration.

In this survey, the initial parts provide a comprehensive summary of the main approaches in the field leveraging the research conducted in English as the most targeted language in every NLP task. This segmentation aims to establish a foundational understanding of STS before the detailed exploration of the specific advancements made in Arabic STS.

Moreover, this sequential presentation allows for establishing a deep comparative analysis and gives valuable insights into the distinct characteristics and advancements within Arabic STS, facilitating an assessment of the progress and trends in Arabic STS in relation to the more extensively documented English-language research and then offering a nuanced perspective on the state of the field in the Arabic context.

The rest of this paper is structured as follows: In Section 2, we delve into the foundational aspects of STS. Section 3 is dedicated to the exploration of STS in the context of the Arabic language and discusses potential future directions in this field. Finally, we present our conclusions in Section 4.

## 2 Semantic Textual Similarity Background

In this section, we provide an in-depth examination of STS and its associated tasks, and we present the different approaches that researchers have developed to measure semantic similarity. These approaches encompass a wide spectrum of techniques, ranging from traditional methods to state-of-the-art deep learning models.

### 2.1 Semantic Textual Similarity and Related Tasks

In NLP, the concept of semantic similarity can be explored across multiple dimension. The basic case is when we study the similarity between words in terms of their meaning. Measuring the similarity between two sentences is more challenging, it needs to extend the analysis beyond individual words and take into account the overall meaning and coherence of the sentences.

More generally, semantic similarity can be also considered at the document level. One of the valuable applications of this case is plagiarism detection which helps users locate documents with related or similar content and maintain the integrity of academic and professional writing.

STS task made its debut at SemEval 2012. The task objective was to automatically assess the semantic similarity between a pair of texts using a predefined scale, such as a range from 0 (not similar) to 5 (completely equivalent) [10]. As Semantic similarity varies across a spectrum, ranging from stark dissimilarity to precise semantic equivalence, employing a graded similarity scale becomes essential to accurately reflect the level of similarity between text pairs.

Furthermore, the research in STS has led to a set of variants and special cases, Short Text Similarity Semantic (STSS) and Semantic Question Similarity are some of these sub-tasks (e.g., [11, 12]). Semantic Short Text Similarity (STSS) is a specialized case of STS that measures semantic similarity between short text fragments. This task is particularly relevant for applications like query reformulation, duplicate detection, and social media analysis, where users express their information needs or opinions in brief, concise formats (e.g., [13]).

Question Similarity can be considered as a case of SSTS where the task focuses on measuring the similarity between questions. This task is essential for various applications, including question-answering systems, community question-

**Table 1.** Example of semantic relatedness and STS

| Text 1 | Text 2 |
|---|---|
| *Darwin's theory of evolution* | *Mendel's laws of inheritance* |
| *A child reading a book* | *A child engrossed in a book* |

answering platforms, and chatbots, where the goal is to identify questions that share a similar meaning or intent.

Another case is Cross-Lingual STS [14]. In this case, the goal is to measure the semantic similarity between texts in different languages. While STS methods focus on nuances within a single language, Cross-Lingual STS tackles the added complexity of variations in language, structure, and cultural context across different languages.

STS finds applications in various specific domains, including clinical texts (e.g., [15, 16]), legal domain (e.g., [17, 18]), and scientific literature (e.g., [19]). While general-purpose STS measures may not perform effectively in certain specialized domains, domain-specific STS measures have been developed. In these specific domains, STS enhances efficiency and accuracy by assisting in content selection, summarization, and information retrieval, ultimately serving as a valuable tool for professionals in each field.

For instance, Clinical STS can assist in identifying relevant patient records, extracting essential information from unstructured medical documents, and generating concise, accurate summaries of patient histories which can help in clinical decision-making (e.g., [16]).

Another variant of STS is Interpretable Semantic Textual Similarity (iSTS) which aims to develop STS systems with the ability to understand and explain the similarity between texts in a clear and interpretable manner.

Recently, Deshpande et al. [20] introduced a new task named Conditional STS (C-STS) which assesses sentence similarity based on a feature described in natural language, referred to as a condition. C-STS aims to reduce the subjectivity and ambiguity of STS, allowing for nuanced language model evaluation across various natural language conditions.

Numerous other NLP tasks share similarities with STS, and although they tackle distinct challenges, it is notable that the methods employed in the literature largely remain consistent across these tasks. It's conceivable that the closest task to STS is Semantic Textual Relatedness. The key difference between these two concepts is that semantic textual relatedness assesses the level of association or connection between texts, while STS quantifies the degree of similarity in meaning or content between texts.

To provide further clarity regarding this distinction, we examine the two examples presented in Table 1. In the first example, the two texts present a degree of relatedness without sharing substantial content or meaning, whereas in the second case, the texts express a common idea with slight linguistic variances, thereby there is a high degree of similarity between these texts.

The tasks of Recognizing Textual Entailment, Machine Translation Evaluation, and Paraphrase Detection exhibit also some similarities with STS. While all of these NLP tasks deal with understanding and comparing text, their specific objectives and applications vary. STS measures text similarity, Machine Translation Evaluation assesses translation quality, Textual Inference evaluates logical relationships, and Paraphrase Detection identifies rephrased content, each contributing to the broader landscape of text analysis and language understanding.

### 2.2 State-of-the-art Approaches for Semantic Similarity

This subsection gives a summary of the main methods used to deal with the STS for English.

#### 2.2.1 Knowledge-based Semantic-similarity Methods

Knowledge-based methods for STS use structured knowledge sources, such as ontologies, knowledge graphs, taxonomies, or dictionaries, to quantify the semantic similarity between texts. These approaches leverage the information contained in knowledge bases, including relationships between concepts as well as their definitions.

Among the most frequently employed knowledge bases in STS measurements are WordNet [21], Wikipedia, and BabelNet [22].

Knowledge-based semantic similarity methods can use the distance between nodes (concepts) within an ontology to measure the similarity, a technique commonly referred to as the path length or edge-counting method. In this approach, the shorter the distance between two concepts in the ontology, the greater their perceived similarity.

Another path length method assesses similarity based on the proximity of their Least Common Subsumer, which is defined as their closest common ancestor. For instance, the Li similarity measure proposed by Li et al. [23] combines both of these preceding techniques.

In addition to path length methods, feature-based and information content methods are also employed. feature-based methods (e.g., [24]) are used to assess semantic similarity by comparing the attributes or features of concepts. This approach involves examining shared characteristics between concepts to determine their similarity.

The information content is a measure of the amount of information or specificity associated with a word or concept in a given knowledge base. Highly specific words are associated with a low information content value, whereas more general and frequently occurring words have higher information content [25]. The semantic similarity between two concepts is determined by the amount of information shared by these concepts.

Whilst in feature-based measures, terms are represented as sets of features, an example of work applying this approach is [24]. This approach involves examining shared characteristics between concepts to determine their similarity. The degree of similarity is proportional to the number of common features they share and inversely related to the presence of distinctive features [26].

#### 2.2.2 Feature Engineering and Handcrafted Representations

Supervised approaches with feature engineering in NLP tasks are based on transforming the input text to a vector of informative features that capture relevant linguistic patterns and semantic information. Through a set of experimentation and fine-tuning of feature engineering techniques, supervised NLP classification models can effectively learn to make good predictions based on the engineered features. These approaches are very well explored in many NLP tasks for several languages. They prove particularly advantageous when working with limited data resources. In the context of STS, these features may capture lexical, syntactic, or semantic information.

#### 2.2.3 Deep Learning Paradigms in STS

Deep neural network-based methods have proven effective across various NLP tasks, especially in the case of STS. In this context, various deep learning models and architectures are explored to capture complex semantic features, facilitating precise quantification of textual similarity.

**Recurrent neural networks and Siamese architectures:** GRUs, LSTM, Bi-LSTM networks, and their variants have emerged as powerful tools in the realm of NLP tasks, particularly in the context of STS. These models with their ability to represent the sequential structure within textual data, yield enriched representations that effectively encapsulate the underlying meaning. These enriched representations are used to capture the similarity between texts.

Additionally, attention mechanisms have been integrated with RNNs to enable the networks to focus on relevant parts of the input, thereby improving their ability to discern subtle semantic nuances. Moreover, attention-based architectures used with RNN further enhance accuracy by focusing on relevant parts of the sentences. RNNs and their variants are also frequently combined with other neural network architectures like Convolutional Neural Networks (CNNs) within the same model to deal with STS (e.g., [27]).

Siamese neural networks are an architecture containing typically two identical sub-networks. Siamese networks aim to capture a meaningful representation of input pairs, ensuring that similar inputs are positioned closely to each other in the learned feature space. This style of architecture is suitable for tasks related to similarity measurement between two comparable things. This architecture is widely used in the literature for the task of STS (e.g., [28, 29, 30]).

**Word and Sentence Embedding:** In NLP, the words are commonly presented as continuous vectors in a multi-dimensional space, where words

with similar meanings or contexts are located closer to each other. This representation captures semantic relationships between words. The early versions of word embedding were static like Word2Vec [31] or GloVe [32] each word or token in the vocabulary is associated with a pre-computed vector representation, and these vectors do not get updated or fine-tuned as the model learns from task-specific data or in the context.

Unlike traditional word embedding contextualized word embeddings, which are context-sensitive, provide different embeddings for a word depending on its context in a given sentence or document. This allows to capture the meaning of a word in a specific context within a sentence or text. Two prominent examples of models that generate contextualized word embeddings are ELMo [33] which uses a Bi-LSTM network to compute word embeddings based on the entire sentence's context.

ELMo embeddings capture syntactic and semantic nuances, making them valuable for various NLP tasks. Building on this foundation, BERT [34], a transformer-based model, further improved contextualized word embeddings.

It employs a masked language modeling objective to understand words in their surrounding context. BERT embeddings have gained widespread popularity and have been fine-tuned for numerous downstream NLP tasks, achieving state-of-the-art results.

Sentence embedding models have proven to perform well on various STS benchmarks, many models have been developed such as InferSent [35], which is trained on the SNLI dataset for natural language inference and is shown to perform well on STS. Another example of sentence embedding is the "Universal Sentence Encoder" [36], which has shown its effectiveness on a wide range of NLP tasks, including STS. Another recent sentence embedding mode is SimCSE [37]. The key idea behind it is to pre-train sentence embeddings in a contrastive learning framework, where semantically similar sentences are brought close together in the embedding space.

**Transformer-based Models:** The core concept of the Transformer architecture [38] is to substitute conventional recurrent and convolutional layers with a self-attention mechanism. This mechanism enables the model to prioritize different segments of the input sequence dynamically, enhancing its predictive capabilities. Self-attention is the key concept of the Transformer architecture. This mechanism calculates attention scores for each word/token in the input sequence based on its relationships with other words.

It allows the model to effectively capture long-range dependencies and contextual information. The computation in Transformers can be highly parallelizable, making them computationally efficient. This contrasts with RNNs, where computations are sequential, limiting their ability to utilize modern hardware effectively. Since the self-attention mechanism lacks inherent positional information, Transformers incorporate positional encodings to provide the model with information regarding the sequence of words in the input.

Transformer-based models like BERT [34] and RoBERTa [39], T5[40], and GPT [41] and their variants have achieved remarkable results in various NLP, particularly on STS. These models capture contextual word meanings and encode sentences into informative vectors. Sentence similarity can be measured by the cosine distance between these vector representations. Fine-tuning these models on STS datasets or using them as feature extractors can yield excellent results.

As a result, large language models like GPT-4 have become valuable tools offering enhanced performance in many NLP tasks including STS applications. The continuous advancements in transformer-based architectures contribute significantly to the improvement of STS models.

### 2.2.4 Cross-lingual Approaches

To tackle cross-lingual semantic similarity, two main methods are commonly employed: Machine Translation approaches and shared semantic space approaches.

Machine Translation (MT) is widely used to address the cross-lingual STS task. This involves converting the two texts under comparison into the same language, to apply a monolingual similarity approach. For instance, Tian et al. [42] proposed a method to deal with STS for Arabic-English, Spanish-English, and Turkish-English sentence pairs.

Their method relies on Machine Translation to convert the sentences into English. Subsequently,

they apply a hybrid approach that combines both supervised learning and deep learning techniques to establish a semantic similarity measure. They used a set of features as input for the supervised machine learning module, including MT evaluation metrics, along with classical similarity features.

The limitation of MT-dependent approaches is their inapplicability to under-resourced languages lacking an efficient MT system. Therefore, some studies have introduced semantic similarity models across different languages that do not rely on Machine Translation.

To address this, various works have proposed methods based on a shared semantic space approach for different languages. Those methods articulate on representing words from different languages in a shared embedding space by training monolingual semantic representations independently of each other, then using a translation matrix, projecting word vector representations of one language into the representation space of the other language.

The translation matrix is computed using a small set of word pairs consisting of words of one language and their translation in the other language. subsequently, the similarity between words of each sentence is obtained by using traditional metrics such as the cosine similarity of their vectors within the shared embedding space. Monolingual approaches are then applied to compute the similarity between sentences (e.g., [43, 44]).

#### 2.2.5 Hybrid Approaches

Hybrid methods for STS represent an integration of multiple techniques for similarity measurements between pieces of text. Typically, hybrid methods incorporate handcrafted features derived from linguistic and semantic analysis, to capture specific linguistic phenomena. Simultaneously, they leverage the power of machine learning algorithms, including deep neural networks, to autonomously learn complex patterns and representations from large datasets.

This combination enables hybrid models to capitalize on the interpretability of rule-based systems and the capacity of machine learning models to discern complex relationships. By merging these diverse methodologies, hybrid approaches aim to overcome the limitations of individual techniques and capitalize on the strengths of each.

As an example of studies using hybrid measures in Semantic Similarity, Panchenko and Morozova [45] combined a set of knowledge-based measures using Wordnet, corpus-based measures using the web, and different classical corpora, in addition to dictionary-based measures using glosses from Wikitionary, Wordnet, and Wikipedia, all combined with supervised learning to achieve better performance.

Moreover, Rychalska et al. [46] proposed a hybrid textual similarity model, incorporating recursive auto-encoders along with penalty or reward scores derived from WordNet. This model was combined with other similarity models in an ensemble to boost its performance.

## 3 Arabic Semantic Textual Similarity

In this section, we discuss the challenges related to the Arabic language, we explore the various approaches and datasets employed in the literature for measuring semantic similarity in Arabic text. Furthermore, we provide insights into potential future directions to advance research in this domain.

### 3.1 Arabic Language and its Challenges

The Arabic language consists of multiple variants. The Modern Standard Arabic (MSA), is regarded as the standardized form and serves as the official language variant and written standard across all Arab nations. Moreover, it predominantly serves as the primary mode of communication for public speaking, media, and education. MSA presents many challenges for NLP due to many levels of ambiguity.

These challenges are well-studied in the literature. In the next, we present a summary of some of these challenges. A more detailed presentation of the morphological and syntactic challenges of MSA can be found in [47] and [48]. One of the difficulties associated with MSA is the lack of diacritics. This difficulty is more complicated when it is associated with the inflectional nature of the Arabic language.

**Table 2.** Example of sentence ambiguity due to the absence of diacritics

| Arabic sentence | English Translation |
|---|---|
| جاء يومَ العيد<br>*jA' ywma AlEyd* | He came on the day of Eid |
| جاء يومُ العيد<br>*jA' ywmu AlEyd* | The day of Eid came |

**Table 3.** Example illustrating the pro-drop property of the Arabic language

| Arabic sentence | Literal translation | English translation |
|---|---|---|
| ساعد غيرك، يساعدك<br>sAEd gyrk, ysAEdk | Help others, help you | If you help another, he helps you |

**Table 4.** Numerous clitic items

| Arabic sentence | English Translation |
|---|---|
| والي | Ruler |
| و+ال+ي<br>w+Al+y | And to me |
| و+ألي<br>w+Oly | And I follow |
| و+آل+ي<br>w+|l+y | And my clan |
| و+آلي<br>w+|ly | And automatic |

For instance, the following example illustrates a sentence's ambiguity arising from the absence of diacritics:

Another challenge faced by Arabic is the absence of capitalization which makes for instance the task of named entity recognition more difficult than in English.

Additionally, the task of sentence boundary detection is more challenging in Arabic since texts written in Arabic do not follow strict punctuation rules [47].

Another characteristic of Arabic that complicates automatic processing is the pro-drop property of the Arabic language. (i.e. a language that allows the omission of certain pronouns when they can be inferred from context) as demonstrated by the following example:

MSA exhibits high inflectional complexity due to its rich system of concatenation, which significantly complicates morphological analysis. Arabic words are structured around roots rather than stems [49].

Proclitics in linguistics are clitics that come before a word, resembling prefixes, such as the Arabic conjunction و (w) meaning 'and' or the definite article ال (Al) meaning 'the'. On the other hand, enclitics are clitics that follow the word, akin to suffixes, like the Arabic object pronoun هم (hm) meaning 'them'.

Arabic allows for multiple affixes and clitics within a single word. For instance, the word وسيكتبونها (wsyktbwnhA) contains two proclitics, one circumfix, and one enclitic [48]. For example (cf. Table 4), the word والي can be analyzed in five different ways [48]. Each of these cases has a different discretization.

Arabic NLP is still an underdeveloped field and then it suffers from a lack of open-source libraries, sufficient resources, and large corpora needed generally for many tasks on NLP and especially on STS.

Despite that the MSA is considered the official variant for all Arab countries. The language used in everyday communication in the Arab world is local dialects. These dialects pose numerous challenges due to their rich linguistic diversity and significant variation from MSA.

This variation encompasses differences in vocabulary, grammar, pronunciation, and even script, all compounded by the scarcity of annotated data. Additional complexities arise from code-switching and the absence of standardized norms. For example, in dialectal Arabic, there aren't consistent standard rules for vocabulary and spelling in written form.

### 3.2 Approaches Applied for Semantic Textual Similarity in Arabic

In this subsection, we explore the various approaches for measuring semantic similarity in Arabic text, including lexical, distributional, and deep learning-based methods. We discuss the

strengths and limitations of each approach. Our objective is to provide a thorough and insightful overview of the present research landscape in Arabic STS, with a focus on the most relevant contributions and the latest advancements in the field.

#### 3.2.1 Feature Engineering and Static Word Embedding with Similarity Measures

An early work that addressed sentence similarity was presented by Wali et al. [50]. They used data created using the word definition extracted from a collection of dictionaries designed for human users. The designed features cover lexical, semantic, and syntactic-semantic levels.

Furthermore, Hammad et al. [51] employed a supervised machine learning approach, incorporating feature engineering that encompasses morphological, semantic, and lexical-based features, to deal with the task of Semantic Question Similarity. Various machine learning algorithms, including SVM and AdaBoost, were experimented with. The Mowdoo3 dataset was employed as the experimental dataset.

Additionally, building on their prior work, the same authors in another study [52] opted for a classical machine learning approach by focusing on feature engineering of semantic, lexical, word embedding, morphological, word-level, and character-level features.

These features are designed to capture various aspects of textual information. The authors employed the XGBoost algorithm within a supervised machine learning framework, leveraging its robustness and effectiveness in handling complex feature sets. This combination of feature-rich representations and XGBoost's modeling prowess constitutes the core methodology employed in their study.

Correspondingly, to address the task of Semantic Question Similarity, Lichouri et al. [53] used a collection of n-gram features and lexical features and employed a variety of classifiers. Sharifi et al. [54] conducted their experiments using among others a set of shallow lexical similarity, word embedding, sentence embedding, word mover's distance, and POS tag overlap.

The approach of static word embedding and similarity measures involves a set of measures applied to the vector representations of the input texts. The vectorization is made by replacing each token with its word embedding vector. Then the complete input text is represented as a sum of these vectors or a weighted sum of these vectors.

Several similarity measures such as cosine similarity, and Euclidean distance, were explored. An example using this approach is the work of Ferrero et al. [55]. They used a CBOW word representation for the Arabic model proposed by Zahran et al. [56].

The approach is based on summing the representation of the words of the sentence without weight or with weights depending on POS and the Inverse Document Frequency. The system outputs a float number ranging from ”0” (representing complete independence of sentence meanings) to ”1” (signifying meaning equivalence).

To conduct their experiments, they used a dataset comprising 750 pairs of sentences drawn from publicly Microsoft Research Video Description Corpus (MSR-Video) (MSRvideo, 2016), which were then manually translated into Arabic.

#### 3.2.2 STS based on Siamese Neural Networks

Siamese neural networks find application in various NLP tasks, including paraphrase identification, and textual entailment, and have also been employed in several prior works for STS, especially for English [29], [57]. Furthermore, they are valuable for tasks involving comparisons between two inputs, extending their utility beyond NLP to domains like facial comparison, image retrieval, and visual object tracking [58].

These type of neural networks find their applications also in Arabic STS, as evidenced by the works of Hammad et al. [52], Othman et al. [59], Einea and Elnagar [60], and Lichouri et al. [53]. These works are elaborated upon below.

Hammad et al. [52] addressed the task of Semantic Question Similarity using a deep learning-based approach, specifically utilizing a Siamese-based recurrent architecture (bi-directional LSTM) trained with pre-defined features and a pre-trained deep bidirectional transformer based on the BERT model. The task is cast as a binary classification, distinguishing between similar and not similar pairs.

**Table 5.** Available Arabic STS datasets

| Dataset | Year | Type | Language | Size | Scale |
|---|---|---|---|---|---|
| AWSS | 2012 | Word Semantic Similarity | MSA | 70 pairs | [0,4] |
| SemEval 2016-task 3 (CQA-MD) | 2016 | Question Semantic Similarity | Arabic | 1,531 questions and 45,164 related question/answer pairs | (Direct, Related, Irrelevant) |
| SemEval 2017-task 1 | 2017 | Sentence Semantic Similarity / Cross-Lingual STS | MSA-English | 2,435 pairs: 2185 for training and 250 for evaluation | [0,5] |
| SemEval 2017-task 1 | 2017 | Sentence Semantic Similarity | MSA | 1,354 pairs: 1104 for training and 250 for evaluation | [0,5] |
| Mawdoo3 Q2Q | 2019 | Question Semantic Similarity | MSA | 15,712 pairs: 11997 for training and 3715 for evaluation | (0,1) |
| ASSD | 2021 | Sentence Semantic Similarity | MSA | 887 pairs | [0,1] |
| Datasets presented by Al Sulaiman et al. [70] | 2022 | Sentence Semantic Similarity | MSA | 1,379 pairs | [0,5] |
| | | | Egyptian Arabic | 1,379 pairs for training and 250 for evaluation | [0,5] |
| | | | Saudi Arabic | 1,379 pairs for training and 250 for evaluation | [0,5] |
| SemEval 2022-task 8 | 2022 | Document Semantic Similarity | MSA | 572 article pairs: 274 for training and 298 for evaluation | [1,4] |

**Table 6.** Results comparison on identical dataset and evaluation settings

| Study | F-score |
|---|---|
| Fadel et al. [63] | 96.499 % |
| | |
| Al-Theiabat and Al-Sadi [65] | 95.924% |
| Al-Bataineh et al. [62] | 93.00% |
| Hammad et al. [52] | 92.99% |
| Sharifi et al. [54] | 82.58% |
| Lichouri et al. [53] | 79.89% |

The Mawdoo3 dataset serves as the dataset for their experiments. Moreover, Othman et al. [59] tackled the task of Semantic Question Similarity in a retrieval setting for both languages English and Arabic.

The approach relies on a deep learning architecture, specifically employing a Siamese-based framework with LSTM enhanced with Attention Mechanism. They also explored the utilization of CNNs incorporated within the Siamese architecture to retrieve pertinent questions. The questions texts are vectorized using Word2Vec CBOW. The evaluation is based on the dataset released by Othman et al. [61] for English. For Arabic, a translation of this same English collection was made using Google Translate with a post-manual verification.

The publicly accessible Quora Question Pairs dataset was employed for training the Siamese LSTM model. The input text is tokenized and vectorized using word2vec embeddings, each comprising 300 dimensions, which were trained on a corpus of 100 billion words. For Arabic, Word2Vec training was conducted using an

English dataset that was translated using Google Translate. Additionally, The work of Einea and Elnagar [60] is also based on a neural network architecture based on Siamese with different types of neural networks based on CNN and RNN. The input text is represented using a static word embedding based on Word2vec trained on two different datasets, namely the Mawdoo3 Q2Q dataset (described later in this paper) for assessing question pairs' similarity, and the Semeval-2016 Task 3 dataset, composed of query questions. Each query question is associated with a list of around 30 question/answer pairs, which vary in their degree of similarity to the query.

#### 3.2.3 Contextualized Word Embedding and Transformers

Contextualized word embeddings are a type of word representation that captures the meaning of a word based on its context within a given sentence or document. In contrast to conventional static word embeddings, which assign a constant vector to each word without considering context, contextualized embeddings dynamically change based on the surrounding words and the overall context in which the word appears. Language models like BERT, ElMo, and GPT leverage the potency of contextualized word embeddings to achieve a heightened understanding of context, empowering them to excel in diverse linguistic tasks.

An instance of employing this method is demonstrated in the research by Al-Bataineh et al. [62] by training Embeddings from Language Models (ELMo) on a text corpus comprising both MSA and dialectic sentences, alongside a fine-grained pairwise similarity layer integrated to enhance the question-to-question similarity model, ensuring accurate predictions across different dialects, even though it has been exclusively trained on question-to-question MSA data.

The Mawdoo3 Q2Q Dataset served as the training data, while the test set was constructed using the MADAR dataset through the extraction of Q2Q pairs specifically focused on Arabic dialects. The training of word embedding is performed using a large dataset aggregated from three diverse sources: Tweets, Arabic Wikipedia, and Mawdoo3 articles.

Furthermore, Hammad et al. [52] utilized the BERT model to generate embeddings for question pairs. By encoding question pairs with BERT to produce a high-dimensional representation that retained semantic nuances.

Using the same dataset as the two previously mentioned works, Fadel et al. [63] performed some operations of data augmentation to enlarge the training dataset size. Subsequently, they used contextualized word embedding to represent the input. The sequence of vectors is fed to a special case of LSTM called ONLSTM [64] with self- attention.

Following the extraction of representations for each question, a function is employed to compute squared distances between vectors representing questions within each pair, facilitating their merging into a single vector. The result is then input into a deep fully connected neural network with a sigmoid output layer to produce the final binary decision.

Al-Theiabat and Al-Sadi [65] experimented using different deep learning models, a CNN-based, an RNN (bi-directional GRU), a multi-head attention network model, and a BERT model. The CNN model begins by encoding words, and subsequently, each question undergoes processing through three successive layers. In each of these layers, a convolutional operation is applied, followed by an activation function, and then max pooling.

Consequently, the output for each question is a feature representation, and the similarity label is determined by calculating the cosine similarity between the features of the two questions.

The input in this case consists of pairs of questions that are merged into one sequence. This sequence is then represented using the dictionary and fed into a bi-directional GRU neural network, which eventually generates the similarity label as the output.

In the case of the BERT model, the approach involved fine-tuning the multilingual model by employing a sentence pairs classification task specifically with Arabic questions.

Saidi et al. [66] investigated the integration of Arabic BERT models in Siamese neural networks to deal with sentence similarity. Their system assigns a discrete similarity score on a scale from 0 to 5, where 0 indicates complete semantic independence and 5 denotes semantic

equivalence. Their system comprises a BERT-based Siamese Network that incorporates contextual embeddings from BERT, the attention mechanism, and the Siamese neural network.

The study explored various Arabic BERT models for embedding input sentences, including AraBERT [67], Arabic-BERT [68], CAMeL-BERT, and the multilingual mBERT, which is capable of handling Arabic texts [69]. The validation of their approach was conducted using Arabic STS datasets from the SEMEVAL 2017 Multilingual STS. The araBERT-based Siamese Network model achieved a Pearson correlation of 0.925 demonstrating the effectiveness of their approach.

In another study by Al Sulaiman et al. [70], transfer learning and knowledge distillation techniques were employed. The authors proposed three strategies for developing STS models for MSA, Egyptian Arabic, and Saudi Arabic. The first strategy involved using automatic machine translation to convert English data from SNLI [71] and MultiNLI [72] datasets into Arabic.

These translated datasets were then used in the fine-tuning stage to adapt Arabic BERT models into STS Arabic ones. The second approach focused on integrating Arabic BERT models with English data sources to enhance Arabic STS models. Lastly, the third approach aimed to improve the performance of knowledge distillation-based models by fine-tuning them with the use of a translated dataset, specifically tailored to enhance their performance in Arabic.

This study encompassed MSA and two Arabic dialects, Egyptian and Saudi Arabian, and proposed valuable datasets through the professional translation of 1.3K sentence pairs from English to MSA, Egyptian Arabic, and Saudi Arabic. The proposed MSA models were evaluated on the SemEval-2017 Arabic evaluation set [73], while the dialect models were tested on a translated version of this dataset crafted by native speakers of both dialects.

### 3.3 Arabic STS Datasets

Despite the presence of numerous publicly accessible STS datasets in English, there is still a considerable deficiency in both the number and size of such datasets in the Arabic language. In the following subsection, we present a set of datasets used in Arabic STS research. Our selection is focused on datasets that are openly accessible to the research community, notably those that have served as evaluation benchmarks for various Arabic STS systems. For STS at the word level, Almarsoomi et al., 2012 [74] have created a benchmark dataset, referred to as the AWSS dataset, which comprises 70 pairs of Arabic words. These pairs have been annotated by 60 native Arabic speakers.

For sentence similarity, in SemEval-2017, Cer et al. [73] provided two distinct datasets. One dataset was to evaluate the Cross-lingual Arabic-English Semantic Similarity, while the other one was for Arabic-Arabic Semantic Similarity. The pairs of sentences were retrieved from diverse English resources. These sentence pairs were subsequently annotated with STS labels and then translated into Arabic. Notably, the translation process was carried out independently from their corresponding pairs.

Translators were provided with an English sentence and its machine-generated Arabic translation, and their task involved correcting any errors before transferring the similarity scores. Another dataset for Arabic sentences was developed by Dahy et al. [75] named the ASSD dataset using collected sentences from Arabic Wikipedia, World Wide Web pages, and The Intermediate Lexicon.

The collected dataset covers different domains. The ASSD dataset underwent a manual evaluation process by seven annotators, who assigned values between 0 and 1 to each sentence pair. Furthermore, Al Sulaiman et al. [70] provided three datasets encompassing sentence pairs translated by experts from the SemEval 2017 English STS dataset into MSA, Egyptian dialect, and Saudi dialects. For Question Semantic Similarity, Seelawi et al. [11] produced the Mawdoo3 Q2Q dataset.

It contains pairs of questions labeled by annotators with 1 if they are semantically similar or 0 otherwise. Another Dataset was used for the study of Question Semantic Similarity, referred to as CQA-MD, was developed by Nakov et al. [76] as part of SemEval 2016, the dataset was released for the Arabic subtask aiming to rank pairs of question and answer, retrieved from Community Question Answering platforms, according to their relevance to a new question.

**Table 7.** Different Arabic STS models based on deep learning or pre-trained models

| | Type of Arabic | Task | Notable Elements in the approach | Word Representation | Transformers | Attention Mechanism | Evaluation data | Results |
|---|---|---|---|---|---|---|---|---|
| Ferrero et al. [55] | MSA | Sentence Similarity | Calculate the cosine similarity between two sentences by summing their weighted or unweighted word vectors. | Pretrained CBOW Word2Vec model | NO | NO | Arabic STS datasets from the SEMEVAL 2017 | Pearson correlation: 76.67% |
| Fadel et al. [63] | MSA | Q2Q | - Ordered Neurons LSTM[64]<br>- Sequence weighted attention | Arabic ELMo | NO | YES | Mawdoo3 Q2Q dataset with Data Augmentation | F1-score: 96.499 % |
| Al-Bataineh et al. [62] | MSA + Arabic Dialect | Q2Q | Several models based on:<br>- Word Embedding layer followed by LSTM or RandLSTM [79], Focus Layer/ Dot Product & Absolute Distance<br>- Sent2Vec, Focus Layer/ Dot Product & Absolute Distance | - Word2Vec (AraVec) and ELMo, trained on Arabic Wikipedia, Mawdoo3, and Twitter<br>- Sent2Vec | NO | NO | Mawdoo3 Q2Q Dataset + MADAR dataset | - Best model on Mawdoo3 Dataset (ELMo + TrainableLSTM + DPAD): F1-score 0.93)<br>- Best model on MADAR dataset (ELMo + TrainableLSTM + FocusLayer): F1-score 0.82 |
| Sharifi et al. [54] | MSA | Q2Q | - Similarity measures using embedding and Word Mover's distance<br>- Doc2vec similarity<br>- POS tag overlap<br>- SVM classifier | FastText | NO | NO | Mawdoo3 Q2Q dataset | F1-score is 82.58% |
| Einea and Elnagar [60] | MSA | Q2Q | - Siamese neural networks with 1D-CNN, BiLSTM, BiGRU<br>- Vector Similarity Layer with Manhattan Distance, Euclidean Distance, and Cosine Distance | A Word2Vec model trained on several datasets | NO | NO | Mawdoo3 Q2Q dataset and SemEval-2016 Task 3 dataset | - Results on NSURL Dataset: Best accuracy 76.9% obtained using 1D-CNN and Euclidean Distance<br>- Results on SemEval Dataset: Accuracy 58.0% |
| Lichouri et al. [53] | MSA | Q2Q | Siamese neural networks | Not indicated | NO | NO | Mawdoo3 Q2Q dataset | F1-score: 79.89% |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Al-Theiabat and Al-Sadi [65] | MSA | Q2Q | Different deep learning models: CNN-based, RNN (bi-directional GRU, Multi-head attention network model, BERT model) | Not indicated | YES | YES | Mawdoo3 Q2Q dataset | The top-performing model is generated by employing an ensemble of pre-trained multilingual BERT models with 95.924% F1-Score |
| Hammad et al. [52] | MSA | Q2Q | - Supervised machine learning with feature engineering using a set of morphological, semantic, and lexical-based features.<br>- Siamese deep learning recurrent architecture<br>- A Pre-trained deep bidirectional transformer based on the BERT model | BERT | YES | NO | Mawdoo3 Q2Q dataset | Best result (BERT-based model) with 92.99% F1-score |
| Al Sulaiman et al. [70] | MSA, Egyptian, and Saudi Arabic dialects | Sentence Similarity | - Tuning Arabic BERT<br>- Combining English STS and Arabic BERT models to develop improved models for Arabic STS<br>- Transfer learning and Knowledge distillation techniques | Several models including ArabicBERT [68] and ARBERT [80] | YES | - | SEMEVAL 2017 Multilingual Semantic Textual Similarity dataset and its manual translation to Egyptian and Saudi Arabic | - Evaluation Results for MSA STS: 81% Spearman rank correlation<br>- Evaluation Results for Dialects: 77.5% Spearman rank correlation for the Egyptian dialect and 76% for the Saudi Arabia dialect |
| Saidi et al. [66] | MSA | Sentence Similarity | Combining BERT, Attention mechanism, and Siamese | BERT. including AraBER, Arabic-BERT, CAMeL-BERT, and the multilingual mBERT | YES | YES | Arabic STS datasets from the SEMEVAL 2017 Multilingual Semantic Textual Similarity | Best Results were obtained on the sub-dataset MSR-Paraphrase dataset with 92.50% Pearson's correlation. |

The data was extracted from three different medical forums.

They used the questions extracted from one forum as original questions, and they associated every question with a set of related question/answer pairs collected from the other two forums using a search engine.

The available training dataset comprises 1,531 original questions associated with 45,164 question/answer pairs annotated with "Direct" if the pair contains a direct answer to the original question, "Related" if it provides a partial answer to the original question, and "Irrelevant" if it doesn't cover any aspects of the original question.

At the document level, to the best of our knowledge, the only available Arabic dataset dedicated especially to document STS is the SemEval-2022 dataset [77] introduced for the news article similarity task.

The data comprises Arabic news article pairs, each annotated with labels ranging from 1 (Very similar) to 4 (Very Dissimilar) regarding seven score categories namely, "Geography," "Entities," "Time," "Narrative," "Overall," "Style," and "Tone". To determine the final label for each article pair, the average of the category scores is calculated. Furthermore, The Open Source Arabic Corpora (OSAC) [78] served as a source corpus for Arabic document similarity studies. The corpus contains 22,429 texts covering various categories (i.e. economics, history, education, health, etc.).

Table 5 describes the key Arabic STS datasets in terms of type, language, size, and the scale used for measuring semantic similarity.

### 3.4 Analysis of Arabic Semantic Textual Similarity and Future Directions

The objective of this subsection is to compare the results of various approaches when applied to Arabic. However, it is important to note that not all of these approaches have been assessed using identical evaluation methods or datasets. Table 7 resumes the different works previously presented. We highlighted the different approaches and the evaluation settings.

This study examines key works from 2017 onwards that pertain to the Arabic language. It specifically concentrates on research employing modern deep learning-based approaches, aiming to identify significant trends and noteworthy developments that have influenced the field during this period.

Table 6 shows a comparison of results from studies that were evaluated on the same dataset (Mawdoo3 Q2Q dataset) and using identical evaluation settings. We observe that the most effective systems utilize word embeddings in conjunction with advanced deep learning architectures, including ONLSTM, LSTM, and the BERT model.

While research in Arabic STS is expanding, certain limitations and challenges persist, impeding the advancement in this field from aligning with that of English. First, the available datasets for Arabic remain limited compared to the vast availability of English datasets. The inadequacy in both the size and number of Arabic datasets poses a significant challenge and negatively impacts the progress of research in this field in comparison to English STS. In NLP tasks, including STS, the availability of large and diverse datasets is crucial for training robust and effective models.

The Arabic language, with its unique characteristics including variations in dialects, introduces additional challenges. Unfortunately, the available STS datasets for Arabic are scarce when it comes to representing the diverse Arabic dialects as well as domain-specific datasets. Addressing these limitations and expanding the range and diversity of Arabic STS datasets is essential for the advancement of research in this field.

Additionally, despite the availability of datasets containing pairs sourced from native Arabic materials, it's noteworthy that some datasets are translations from English datasets. Translation can introduce additional complexities and potential discrepancies in linguistic nuances, cultural references, and idiomatic expressions between the source and target languages.

Consequently, models trained on translated datasets may encounter challenges in capturing the complexity of Arabic language usage and may exhibit biases or limitations in their performance. Thus, while leveraging translated datasets can expand the scope of available data for Arabic NLP tasks, careful consideration of the potential impacts of translation on dataset quality and model performance is essential for robust and reliable results.

On the other hand, Models trained on MSA or one dialect might not work well with others because of differences in words, grammar, and sentence structure. Ensuring the generalizability of STS models across Arabic dialects requires training data that encompasses a wide range of dialectal variations and fine-tuning or adapting models on dialect-specific data to improve performance. Syntactic differences including variations in word order, verb conjugation, and syntactic structures across Arabic dialects can affect the alignment of text pairs in STS tasks.

Arabic dialects differ in terms of idiomatic expressions, cultural references, and ways of expressing ideas, which may not be captured adequately by models trained in Standard Arabic. To enhance model performance, it's essential to incorporate linguistic features specific to Arabic dialects into model architectures, augment training data with dialectal variations and diverse linguistic phenomena, and develop evaluation benchmarks that take into account the dialectal differences and linguistic nuances of the Arabic language.

## 4 Conclusion

The paper provides a comprehensive examination of the STS task. It offers insights into various strategies employed to tackle the STS challenge in English while delving deeply into the specific progressions achieved in Arabic STS. Through comparative analysis, the paper unveils the attributes and advancements within Arabic STS, shedding light on valuable insights and future research directions. Despite strides made in Arabic STS, there remains significant work to be done, both in terms of the approaches used and the available datasets.

In comparison to the English language, there is little dataset-related work. Modern approaches based on deep learning, contextualized word embeddings, attention mechanisms, and vectorized text representations like the BERT language model have been minimally explored in the context of the Arabic language. The application of recent advancements, including Large Language Models (LLMs), is still underdeveloped, highlighting a gap compared to English and emphasizing the need for extensive research and development of comprehensive datasets.

## References

1. **Severyn, A., Moschitti, A. (2015).** Learning to rank short text pairs with convolutional deep neural networks. Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 373–382. DOI: 10.1145/2766462.276773.
2. **Gonzales, A. R., Mascarell, L., Sennrich, R. (2017).** Improving word sense disambiguation in neural machine translation with sense embeddings. Proceedings of the Second Conference on Machine Translation, pp. 11– 19.
3. **Babar, S., Patil, P. D. (2015).** Improving performance of text summarization. Procedia Computer Science, 46, pp. 354-363. DOI: 10.1016/j.procs.2015.02.031.
4. **Risch, J., Möller, T., Gutsch, J., Pietsch, M. (2021).** Semantic answer similarity for evaluating question answering models. DOI: 10.48550/arXiv.2108.06130.
5. **Vrbanec, T., Meštrović, A. (2017).** The struggle with academic plagiarism: Approaches based on semantic similarity. 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE, pp. 870-875. DOI: 10.23919/MIPRO.2017.7973544.
6. **Arabi, H., Akbari, M. (2022).** Improving plagiarism detection in text documents using hybrid weighted similarity. Expert Systems with Applications, Vol. 207, p. 118034. DOI: 10.1016/j.eswa.2022.118034.
7. **Alian, M., Awajan, A. (2018).** Arabic semantic similarity approaches - review. 2018 International Arab Conference on Information Technology (ACIT), IEEE, pp. 1–6. DOI: 10.1109/ACIT.2018.8672665.
8. **Alian, M., Awajan, A. (2020).** Semantic similarity for English and Arabic texts: a review. Journal of Information & Knowledge Management, Vol. 19, No. 4, p. 2050033. DOI: 10.1142/S0219649220500331.
9. **Abo-Elghit, A. H., Al-Zoghby, A. M., Hamza, T. T. (2020).** Textual similarity measurement approaches: A survey (1). Egyptian Journal of Language and Engineering, Vol. 7 No. 2, pp. 41–62. DOI: 10.21608/ejle.2020.42018.1012.
10. **Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A. (2012).** Semeval-2012 task 6: A pilot on semantic textual similarity. SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the

shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pp. 385–393.

11. **Seelawi, H., Mustafa, A., Al-Bataineh, H., Farhan, W., Al-Natsheh, H. T. (2019).** NSURL-2019 task 8: Semantic question similarity in Arabic. Proceedings of the First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) colocated with ICNLSP 2019-Short Papers, pp. 1–8.
12. **Croft, D., Coupland, S., Shell, J., Brown, S. (2013).** A fast and efficient semantic short text similarity metric. 2013 13th UK Workshop on Computational Intelligence, IEEE, pp. 221–227. DOI: 10.1109/UKCI.2013.6651309.
13. **Alnajran, N., Crockett, K., McLean, D., Latham, A. (2018).** An empirical performance evaluation of semantic-based similarity measures in microblogging social media. 2018 IEEE/ACM 5th International Conference on Big Data Computing Applications and Technologies (BDCAT), IEEE, pp. 126–135. DOI: 10.1109/BDCAT.2018.00023.
14. **Agirre, E., Banea, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Mihalcea, R., Wiebe, J. (2016).** Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. SemEval-2016, 10th International Workshop on Semantic Evaluation, pp. 497–511.
15. **Hassanzadeh, H., Nguyen, A., Verspoor, K. (2019).** Quantifying semantic similarity of clinical evidence in the biomedical literature to facilitate related evidence synthesis. Journal of Biomedical Informatics, Vol. 100, p. 103321. DOI: 10.1016/j.jbi.2019.103321.
16. **Wang, Y., Liu, S., Afzal, N., Rastegar-Mojarad, M., Wang, L., Shen, F., Liu, H. (2018).** Overview of the BioCreative/OHNLP challenge 2018 task 2: clinical semantic textual similarity. Proceedings of the BioCreative/ OHNLP Challenge, Vol. 2018, pp. 1–5.
17. **Mandal, A., Chaki, R., Saha, S., Ghosh, K., Pal, A., Ghosh, S. (2017).** Measuring similarity among legal court case documents. Proceedings of the 10th Annual ACM India Compute Conference, pp. 1–9. DOI: 10.1145/ 3140107.314011.
18. **Sugathadasa, K., Ayesha, B., de-Silva, N., Perera, A. S., Jayawardana, V., Lakmal, D., Perera, M. (2017).** Synergistic union of word2vec and lexicon for domain specific semantic similarity. 2017 IEEE international conference on industrial and information systems, pp. 1–6. DOI: 10.1109/ICIINFS.2017. 8300343.
19. **Sellak, H., Ouhbi, B., Frikh, B. (2015).** Using rule-based classifiers in systematic reviews: a semantic class association rules approach. Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services, pp. 1–5. DOI: 10.1145/2837185.283727.
20. **Deshpande, A., Jimenez, C. E., Chen, H., Murahari, V., Graf, V., Rajpurohit, T., Narasimhan, K. (2023).** C-STS: Conditional semantic textual similarity. arXiv preprint arXiv:2305.15093. DOI: 10.48550/arXiv.23 05.15093.
21. **Miller, G. A. (1995).** WordNet: A lexical database for English. Communications of the ACM, Vol. 38, No. 11, pp. 39–41. DOI: 10.11 45/219717.219748.
22. **Navigli, R., Ponzetto, S. P. (2012).** BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial intelligence, Vol. 193, pp. 217–250. DOI: 10.1016/j.artint.2012. 07.001.
23. **Li, Y., Bandar, Z. A., McLean, D. (2003).** An approach for measuring semantic similarity between words using multiple information sources. IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No. 4, pp. 871–882. DOI: 10.1109/TKDE.2003.1209005.
24. **Jiang, Y., Zhang, X., Tang, Y., Nie, R. (2015).** Feature-based approaches to semantic similarity assessment of concepts using Wikipedia. Information Processing & Management, Vol. 51, No. 3, pp. 215–234. DOI: 10.1016/j.ipm.2015.01.001.
25. **Resnik, P. (1995).** Using information content to evaluate semantic similarity in a taxonomy. ArXiv Preprint cmp-lg/9511007. DOI: 10.485 50/arXiv.cmp-lg/9511007.

**26. Tversky, A. (1977).** Features of similarity. Psychological Review, Vol. 84, No. 4, p. 327. DOI: 10.1037/0033-295X.84.4.327.

**27. Tien, N. H., Le, N. M., Tomohiro, Y., Tatsuya, I. (2019).** Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity. Information Processing & Management, Vol. 56, No. 6, p. 102090. DOI: 10.1016/j.ipm.2019.102090.

**28. Ranasinghe, T., Orăsan, C., Mitkov, R. (2019).** Semantic textual similarity with siamese neural networks. Proceedings of the International Conference on Recent Advances in Natural Language Processing, pp. 1004–1011. DOI: 10.26615/978-954-452-056-4_116.

**29. Neculoiu, P., Versteegh, M., Rotaru, M. (2016).** Learning text similarity with siamese recurrent networks. Proceedings of the 1st Workshop on Representation Learning for NLP, pp. 148–157.

**30. Lv, C., Wang, F., Wang, J., Yao, L., Du, X. (2020).** Siamese multiplicative LSTM for semantic text similarity. Proceedings of the 2020 3rd International Conference on Algorithms, Computing and Artificial Intelligence, No. 28, pp. 1–5. DOI: 10.1145/3446132.3446160.

**31. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., Dean, J. (2013).** Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems, pp. 3111–3119.

**32. Pennington, J., Socher, R., Manning, C. D. (2014).** Glove: Global vectors for word representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543.

**33. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L. (2018).** Deep contextualized word representations. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 pp. 2227–2237.

**34. Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2019).** BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1. pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

**35. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A. (2017).** Supervised learning of universal sentence representations from natural language inference data. arXiv preprint arXiv:1705.02364. DOI: 10.48550/arXiv.1705.02364.

**36. Cer, D., Yang, Y., Kong, S. Y., Hua, N., Limtiaco, N., John, R. S., Kurzweil, R. (2018).** Universal sentence encoder for English. Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations, pp. 169–174. DOI: 10.18653/v1/D18-2029.

**37. Gao, T., Yao, X., Chen, D. (2021).** SimCSE: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821. DOI: 10.48550/arXiv.2104.08821.

**38. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017).** Attention is all you need. 31st Conference on Neural Information Processing Systems, Vol. 30.

**39. Liu, Y. (2019).** Roberta: A robustly optimized bert pretraining approach. ArXiv Preprint ArXiv:1907.11692.

**40. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Liu, P. J. (2020).** Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, Vol. 21, No. 140, pp. 1–67.

**41. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. (2018).** Improving language understanding by generative pre-training.

**42. Tian, J., Zhou, Z., Lan, M., Wu, Y. (2017).** ECNU at SemEval-2017 task 1: Leverage Kernel-based traditional NLP features and neural networks to build a universal model for multilingual and cross-lingual semantic textual similarity. Proceedings of the 11th international

workshop on semantic evaluation, pp. 191–197. DOI: 10.18653/v1/S17-2028.

43. **Glavaš, G., Franco-Salvador, M., Ponzetto, S. P., Rosso, P. (2018).** A resource-light method for cross-lingual semantic textual similarity. Knowledge-Based Systems, Vol. 143, pp. 1–9. DOI: 10.1016/j.knosys.2017.11.041
44. **Brychcín, T. (2020).** Linear transformations for cross-lingual semantic textual similarity. Knowledge-Based Systems, Vol. 187, p. 104819. DOI: 10.1016/j.knosys.2019.06.027.
45. **Panchenko, A., Morozova, O. (2012).** A study of hybrid similarity measures for semantic relation extraction. Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data, pp. 10–18.
46. **Rychalska, B., Pakulska, K., Chodorowska, K., Walczak, W., Andruszkiewicz, P. (2016).** Samsung poland NLP team at SemEval-2016 Task 1: Necessity for diversity; combining recursive autoencoders, WordNet and ensemble methods to measure semantic similarity. Proceedings of the 10th International Workshop on Semantic Evaluation, pp. 602–608.
47. **Farghaly, A., Shaalan, K. (2009).** Arabic natural language processing: challenges and solutions. ACM Transactions on Asian Language Information Processing, Vol. 8, No. 4, DOI: 10.1145/1644879.1644881.
48. **Habash, N. Y. (2010).** Introduction to Arabic natural language processing. Synthesis Lectures on Human Language Technologies, Vol. 3, No. 1.
49. **Attia, M., (2008).** Handling Arabic morphological and syntactic ambiguity within the LFG framework with a view to machine translation. Manchester: University of Manchester, Vol. 279.
50. **Wali, W., Gargouri, B., Hamadou, A. B. (2015).** Supervised learning to measure the semantic similarity between Arabic sentences. Computational Collective Intelligence: 7th International Conference, Springer, pp. 158–167. DOI: 10.1007/978-3-319-24069-5_15.
51. **Hammad, M. M., Al-Smadi, M., Baker, Q. B., Al-Asa'd, M., Al-Khdour, N., Younes, M. B., & Khwaileh, E. (2020).** Question to question similarity analysis using morphological, syntactic, semantic, and lexical features. Journal of Universal Computer Science, Vol. 26, No. 6, pp. 671–697.
52. **Hammad, M., Al-Smadi, M., Baker, Q. B., Sa'ad, A. (2021).** Using deep learning models for learning semantic text similarity of Arabic questions. International Journal of Electrical and Computer Engineering, Vol. 11, No. 4, p. 3519. DOI: 10.11591/ijece.v11i4.pp3519-3528.
53. **Lichouri, M., Abbas, M., Benaziz, B., Freihat, A. A. (2019).** ST NSURL 2019 shared task: Semantic question similarity in Arabic. Proceedings of The First International Workshop on NLP Solutions for Under Resourced Languages, pp. 80–84.
54. **Sharifi, A., Hassanpoor, H., Maduyieh, N. Z. (2019).** AtyNegar at NSURL-2019 task 8: Semantic question similarity in Arabic. Proceedings of the First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) colocated with ICNLSP 2019-Short Papers, pp. 31–36.
55. **Ferrero, J., Schwab, D. (2017).** LIM-LIG at SemEval-2017 task1: Enhancing the semantic similarity for Arabic sentences with vectors weighting. Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 134–138. DOI: 10.18653/v1/S17-2017.
56. **Zahran, M. A., Magooda, A., Mahgoub, A. Y., Raafat, H., Rashwan, M., Atyia, A. (2015).** Word representations in vector space and their applications for Arabic. Computational Linguistics and Intelligent Text Processing: 16th International Conference, Springer, pp. 430–443. DOI: 10.1007/978-3-319-18111-0_32.
57. **Mueller, J., Thyagarajan, A. (2016).** Siamese recurrent architectures for learning sentence similarity. Proceedings of the AAAI conference on artificial intelligence Vol. 30, No. 1. DOI: 10.1609/aaai.v30i1.10350.
58. **Ilina, O., Ziyadinov, V., Klenov, N., Tereshonok, M. (2022).** A survey on symmetrical neural network architectures and

applications. Symmetry, Vol. 14, No. 7. p. 1391. DOI: 10.3390/sym14071391.

59. **Othman, N., Faiz, R., Smaïli, K. (2022).** Learning English and Arabic question similarity with Siamese neural networks in community question answering services. Data & Knowledge Engineering, Vol. 138, p. 101962. DOI: 10.1016/j.datak.2021.101962.
60. **Einea, O., Elnagar, A. (2019).** Predicting semantic textual similarity of Arabic question pairs using deep learning. 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), IEEE, pp. 1–5. DOI: 10.1109/AICCSA47632.2019.9035362.
61. **Othman, N., Faiz, R., Smaïli, K. (2019).** Enhancing question retrieval in community question answering using word embeddings. Procedia Computer Science, Vol. 159, pp 485–494. DOI: 10.1016/j.procs.2019.09.203.
62. **Al-Bataineh, H., Farhan, W., Mustafa, A., Seelawi, H., Al-Natsheh, H. T. (2019).** Deep contextualized pairwise semantic similarity for Arabic language questions. 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, pp. 1586–1591. DOI: 10.1109/ICTAI.2019.00229.
63. **Fadel, A., Tuffaha, I., Al-Ayyoub, M. (2019).** Tha3aroon at NSURL-2019 task 8: Semantic question similarity in Arabic. arXiv preprint arXiv:1912.12514. DOI: 10.48550/arXiv.1912.12514.
64. **Shen, Y., Tan, S., Sordoni, A., Courville, A. (2018).** Ordered neurons: Integrating tree structures into recurrent neural networks. ArXiv Preprint, ArXiv181009536. DOI: 10.48550/arXiv.1810.09536.
65. **Al-Theiabat, H., Al-Sadi, A. (2020).** The inception team at NSURL-2019 task 8: Semantic question similarity in Arabic. ArXiv Preprint, ArXiv200411964. DOI: 10.48550/arXiv.2004.11964.
66. **Saidi, R., Jarray, F., Alsuhaibani, M. (2023).** SiameseBERT: A bert-based siamese network enhanced with a soft attention mechanism for Arabic semantic textual similarity. ICAART No. 3, pp. 146–151. DOI: 10.5220/0011624800003393.
67. **Antoun, W., Baly, F., Hajj, H. (2020).** AraBERT: Transformer-based model for Arabic language understanding. ArXiv Prepr, ArXiv200300104. DOI: 10.48550/arXiv.2003.00104.
68. **Safaya, A., Abdullatif, M., Yuret, D. (2020).** KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. ArXiv Prepr, ArXiv200713184. DOI: 10.48550/arXiv.2007.13184.
69. **Libovickỳ, J., Rosa, R., Fraser, A. (2019).** How language-neutral is multilingual BERT? ArXiv Prepr, ArXiv191103310. DOI: 10.48550/arXiv.1911.03310.
70. **Al-Sulaiman, M., Moussa, A. M., Abdou, S., Elgibreen, H., Faisal, M., Rashwan, M. (2022).** Semantic textual similarity for modern standard and dialectal Arabic using transfer learning. PloS one, Vol. 17, No. 8, p. e0272991. DOI: 10.1371/journal.pone.0272991.
71. **Bowman, S. R., Angeli, G., Potts, C., Manning, C. D. (2015).** A large annotated corpus for learning natural language inference. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 632–642. DOI: 10.18653/v1/D15-1075.
72. **Williams, A., Nangia, N., Bowman, S. (2018).** A broad-coverage challenge corpus for sentence understanding through inference. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 1112–1122. DOI: 10.18653/v1/N18-1101.
73. **Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., Specia, L. (2017).** Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. ArXiv Preprint, ArXiv:1708.00055. DOI: 10.48550/arXiv.1708.00055.
74. **Almarsoomi, F. A., O'Shea, J. D., Bandar, Z. A., Crockett, K. A. (2012).** Arabic word semantic similarity. Proceedings of the World Academy of Science, Engineering and Technology. World Academy of Science, Engineering and Technology.

**75. Dahy, B., Farouk, M., Fathy, K. (2021).** ASSD: Arabic Semantic Similarity Dataset. Proceedings of the 2021 9th International Japan-Africa Conference on Electronics, Communications, and Computations (JAC-ECC), IEEE, pp. 130–134. DOI: 10.1109/JAC-ECC54461.2021.9691424.

**76. Nakov, P., Márquez, L., Moschitti, A., Magdy, W., Mubarak, H., Freihat, A. A., Glass, J., Randeree, B. (2016).** SemEval-2016 Task 3: Community Question Answering. In S. Bethard, M. Carpuat, D. Cer, D. Jurgens, P. Nakov, & T. Zesch (Eds.), Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, California: Association for Computational Linguistics, pp. 525–545. DOI: 10.18653/v1/S16-1083.

**77. Chen, X., Zeynali, A., Camargo, C. Q., Flöck, F., Gaffney, D., Grabowicz, P. A., Samory, M. (2022).** SemEval-2022 Task 8: Multilingual news article similarity. pp. 1094-1106. DOI: 10.18653/v1/2022.semeval-1.155.

**78. Saad, M., Ashour, W. (2010).** OSAC: Open source Arabic corpora. 6th International Conference on Electrical and Computer Systems. DOI: 10.13140/2.1.4664.9288.

**79. Wieting, J., Kiela, D. (2019).** No training required: Exploring random encoders for sentence classification. arXiv preprint arXiv:1901.10444. DOI: 10.48550/arXiv.1901.10444.

**80. Abdul-Mageed, M., Elmadany, A., Nagoudi, E. M. B. (2020).** ARBERT & MARBERT: Deep bidirectional transformers for Arabic. arXiv preprint arXiv:2101.01785. DOI: 10.48550/arXiv.2101.01785.

*Article received on 16/06/2024; accepted on 08/07/2024.*
**Corresponding author is Samira Boudaa.*

ISSN 2007-9737

# Detection and Classification of Multiple Sclerosis from Brain MRIs by Using MobileNet 2D-CNN Architecture

Sudhanshu Saurabh[1, 2,*], P. K. Gupta[3]

[1] Jaypee University of Information Technology, Wakanaghat, India

[2] East Point College of Engineering and Technology, Bangalore, India

[3] Mohan Babu University, School of Computing, Department of Data Science, India

ssmiete@gmail.com, pkgupta@ieee.org

**Abstract.** Deep learning-based object detection and classification have been widely investigated for neuroimaging. Magnetic resonance imaging (MRI) data serve as a diagnostic tool for the detection and classification of brain disorders such as Parkinson's, Alzheimer's disease (AD), and Multiple Sclerosis (MS). In addition, the use of the Convolutional Neural Network (CNN) framework helps in the development of predictive models from the available MRI images. This work aims to develop a CNN-based model with a pre-trained MobileNet model to detect and classify multiple sclerosis using the MRI image dataset. In this article, we have proposed a pre-trained MobileNet-2D-CNN architecture for the accurate prediction of multiple sclerosis from various MRI images. Initially, the proposed model extracted images from MRI images of the affected patient with MS and healthy control. We used MRI images to train the MobileNet-2D-CNN model to identify the MS features map that predicts MS. The proposed architecture has been validated on standard MRI scans. We also performed a class activation map for the interpretation of the prediction provided by the proposed model, which represents the behavior of neurons in the early stages. The proposed approach achieves a classification precision of 98.15% and AUC=1.00.

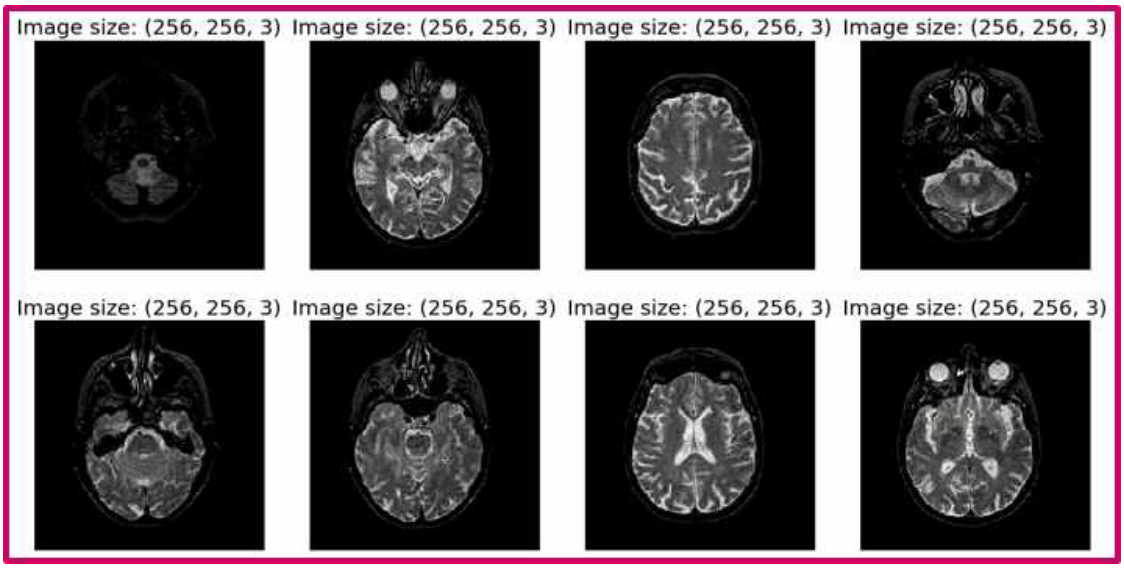
**Keywords.** CNN, deep learning, feature map, mobilenet, MRI, multiple sclerosis.

## 1 Introduction

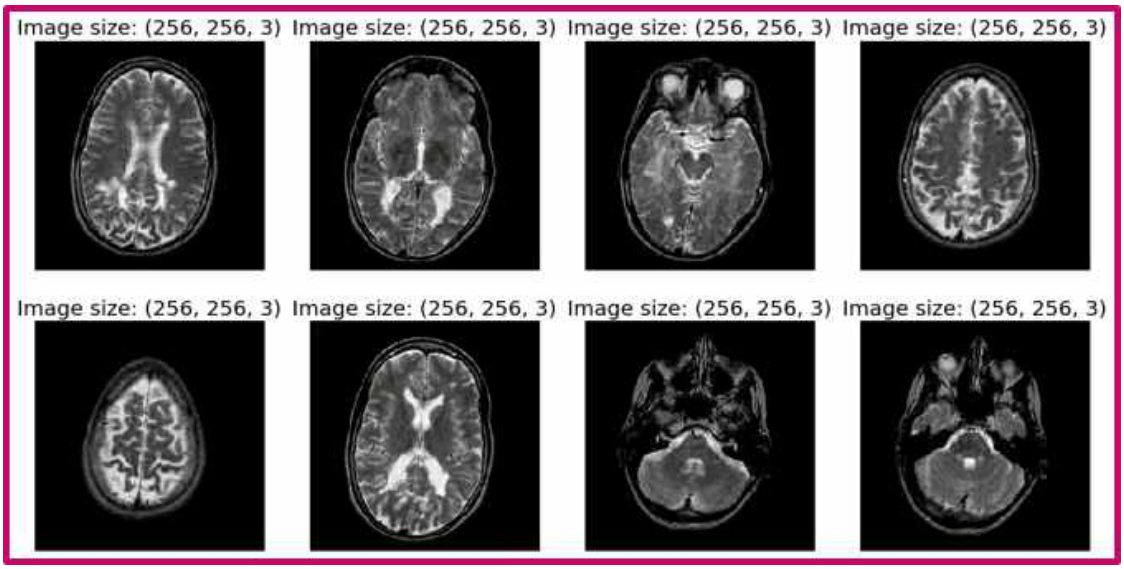
Automatic detection and classification of various brain disorders such as Alzheimer's disease (AD), brain tumors, multiple sclerosis, schizophrenia, and Parkinson's have become a major concern in modern healthcare. Multiple sclerosis (MS) is an irreversible degenerative brain disorder characterized by loss of cognitive function and has no proven cure. MS is a condition and demyelinating disease of the central nervous system that affects mainly the brain and spinal cord and damages the protective covering of nerve fibers resulting in vision loss, depression, decreased sensitivity, mobility problems, and several other issues related to thinking, learning, and planning [3, 32]. The progression of MS can be classified into two subsequent classes known as Benign and Malignant. These two terms represent the severity of MS, where benign MS represents a mild course of multiple sclerosis and malignant MS represents a significant level of disability in various patients. To examine the extent of psychometric properties, the Expanded Disability Status Scale (EDSS) is used as an indicator for patients affected

ISSN 2007-9737






(a)




(b)

**Fig. 1.** Sample image of brain MRI scan dataset (a) Normal healthy control (b) Multiple sclerosis (MS)

by MS [30]. Saliency maps are applied to healthy and Alzheimer's fMRI images by the authors in [25].

Among the various medical imaging techniques, magnetic resonance imaging is one of the most efficient and suitable neuroimaging methods for detecting the presence of white matter lesions in the brain that suggest MS and also effectively diagnose the abnormality in the brain region due to MS [4, 18]. The appearance and shape of the lesions can vary during the MRI scan at each time point [4, 5]. Moreover, measuring the growth of the lesion by using the white signals visible in various MRI scans is another challenging task, as in some MRI scans the lesion disappears at a single time

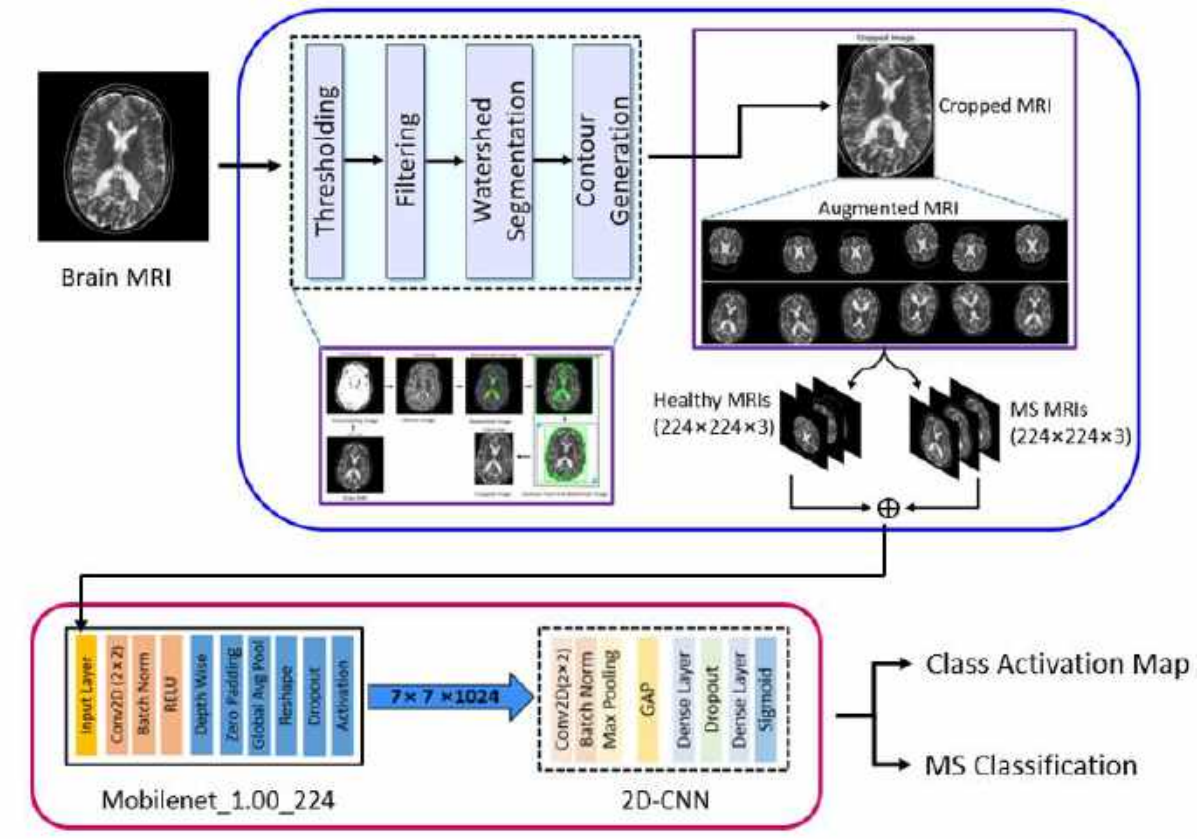



**Fig. 2.** Proposed architecture using brain MRI images for multiple sclerosis detection and classification

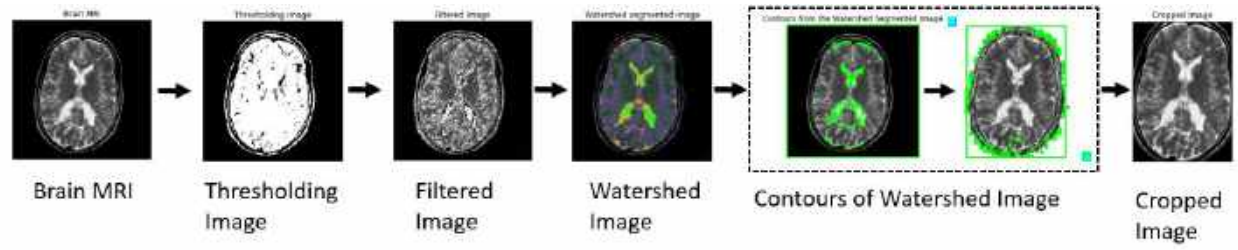
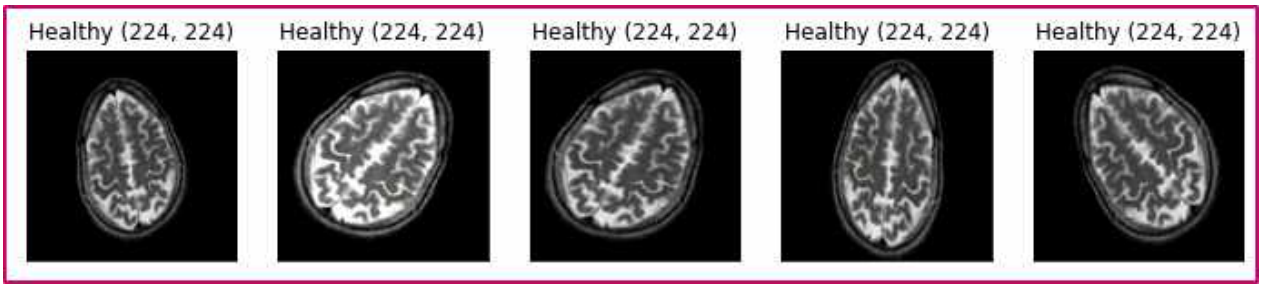


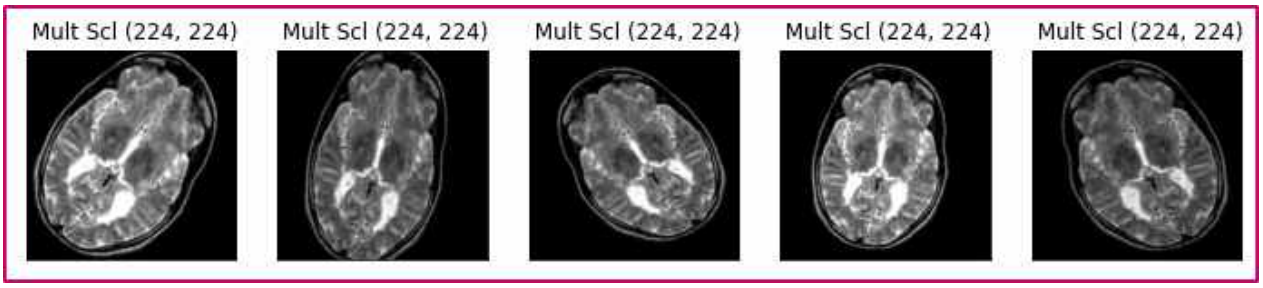

**Fig. 3.** Preprocessing steps involve cropping the brain MRI image

point or lesion time point (LTP) [12]. As shown in Fig. 1, magnetic resonance imaging is used to detect the presence of a white matter lesion in the specific region of the brain to find the MS lesion.

Through experimental studies, deep learning networks such as LiviaNet, HyperDenseNet, and the Convolutional Neural Network (CNN) are widely used for the detection and classification of various neural syndromes [1, 8]. In this work, an experiment is presented to perform a two-dimensional CNN (2D-CNN) model that selects the slices and training input. In the first case, we perform numerical and gradient-based learning at each stage of the densely connected network and are treated as a feature extraction engine that processes a set of effective features




(a)




(b)

**Fig. 4.** Augmented image of brain MRI scan dataset (a) Augmented healthy brain MRI (b) Augmented multiple sclerosis brain MRI

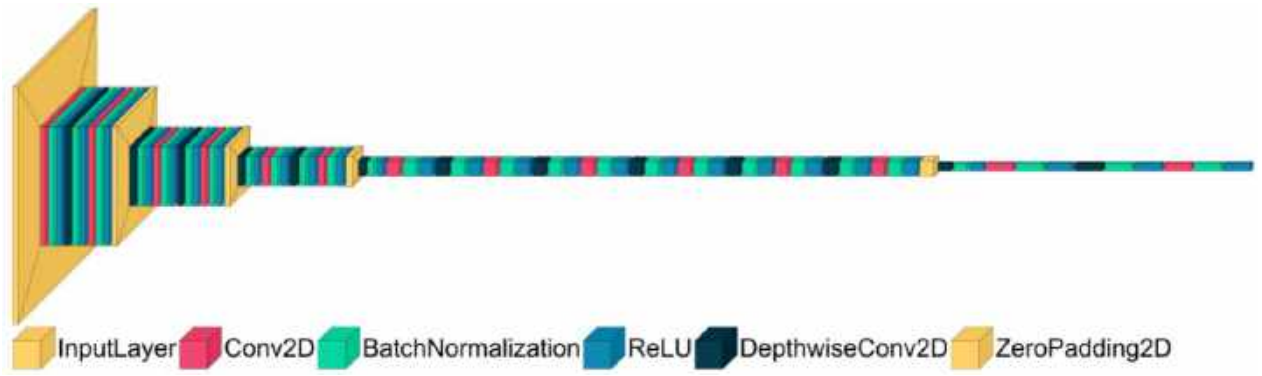



**Fig. 5.** MobileNet architecture

of the MRI dataset. In this study, we have demonstrated the feature extraction of MRI images from the proposed MobileNet-2D-CNN architecture. The size of the MobileNet network is small, which reduces the computation time and reduces the problem of overfitting [17, 29].

Once the feature extraction process is completed, then fed to the 2D-CNN which constructs the feature map from the input segment through the convolutional layer, the max pooling layer, and the fully connected layer handles various 3D-MRI images for detection and classification of MS. The main highlights of this research study are as follows:

– Detect and classify the MS lesion from the MRI images by using the deep learning method.

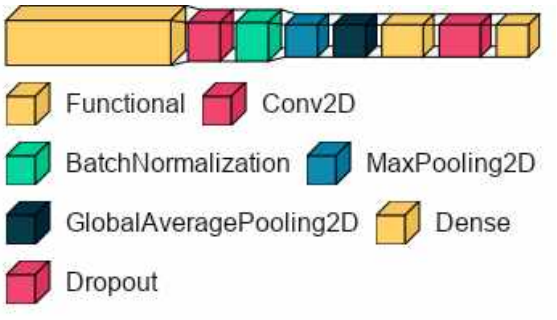



**Fig. 6.** Proposed MobileNet-2D-CNN model

– The proposed model trains the network using the MRI images that show the high signal in the periarterial white matter on the T2 weighted image data.
– Generates the class activation map, which provides the visual diagnosis and classification of MS lesions.

## 2 Related Work

Various studies have been conducted with essential findings that discriminate MS disease from healthy control [18]. These studies focus on the use of machine learning and deep learning-based frameworks and consider MRI scans to find the severity of the disease.

The authors of [4] have considered a CNN architecture based on deep learning and used a vast volume of brain MRI data voxel data for further analysis of MS lesion segmentation. Using 4D-fMRI images, the authors of [26] developed a deep learning model and a reshaping algorithm for the classification of ADHD conditions.

The authors in [20]have proposed a CNN-based MS classifier known as DeepScan for the identification of MS lesions. In [1], the authors have discussed the various deep learning-based techniques that consider the publicly available Brain MRI image segmentation dataset.

In [2], authors have discussed the framework known as an ITK toolkit and applied the watershed transform method for the detection of region boundaries in MRI scans. In [27], the authors have focused on effective neural network training to classify MRI images with limited data and to prevent overfitting the network.

They have simply adopted the proposed model to increase the size of the dataset by using image data augmentation that improves accuracy. In [17], the authors have used precision reduction methods to reduce spatial resolution in different features of the feature map generated by the LeNet-5 network.

The authors in [22, 23], discussed the discrimination of the region in the input image, and the class activation map is influenced by the deep learning model for classification. The authors of [10, 11] have discussed various predictive frameworks and algorithms for the classification and detection of various diseases.

## 3 Materials and Methods

### 3.1 Data Analysis

The experimental MRI dataset used in this study consists of unlabeled brain images from MRI scans to detect and classify MS injury from the healthy control.

This data set is acquired from the publicly available Whole Brain Atlas[1]. This data set consists of standard slices and axial orientations of various MRI scans of the brain. Furthermore, the abnormalities appearing in the data set are based on pre-arterial white matter, pre-gadolinium, and post-gadolinium in the T1 and T2-weighted MRI images, respectively.

[1] www.med.harvard.edu/AANLIB/

**Table 1.** Proposed architecture of MobileNet-2D-CNN

| OPERATION | DATA DIM | WEIGHTS(N) |
|---|---|---|
| Input Layer | $224 \times 224 \times 3$ | – |
| MobileNet_1.00_224 | $7 \times 7 \times 1024$ | 3228864 |
| Conv2D | $5 \times 5 \times 32$ | 294944 |
| Batch Normalization | $5 \times 5 \times 32$ | 128 |
| Max Pooling 2D | $1 \times 1 \times 32$ | – |
| Global Avg Pooling 2D | 32 | – |
| Dense | – | 8448 |
| ReLU | 256 | – |
| Dropout | 256 | – |
| Dense | – | 257 |
| Sigmoid | 1 | – |

This data set also consists of distinct 4D images of brain magnetic resonance imaging scans of 30-year-old patients acquired with axial T1-weighted data. These brain magnetic resonance images show a high signal in white matter in the T2-weighted images.

Here, the MRI images obtained from the normal control and MS subjects are four-dimensional (4D) brain images. These 4D images are defined as ($256 \times 256 \times 22 \times 3$) along with the dimension of the pixel ($1 \times 1 \times 1$) $mm^3$ and the size of the voxel 3 $mm^3$. The 4D MRI image data with 22 slices and a voxel size of 3 $mm^3$ are isotropic for the detection and classification of MS from the normal control. The ratio of MRI images of patients affected with MS with healthy subjects is 22:52.

### 3.2 Methodology

In this section, we have described the proposed model for the detection and classification of MS lesions, as shown in Fig. 2.

The proposed model consists of three modules, i.e. Image preprocessing, MobileNet, and 2D-CNN. The detailed functioning of these modules is discussed in the following subsections. The architecture of the pre-trained MobileNet convolutional neural network is used for the initial stage of dataset training, the MobileNet contains a stack of layer blocks named as convolutional layer, Batch Normalization, Depthwise convolutional layer followed by Global Average Pooling (GAP) layer. Using the MobileNet to classify the natural images is not similar to the numerical data array, so more convolutional blocks are required to properly tune the brain MRI image dataset.

#### 3.2.1 MRI Pre-processing

Here, we apply various preprocessing steps, such as adjusting, readjusting, and enhancing, to the MRI images before feeding them into the MobileNet-2D-CNN model. These preprocessing steps are discussed as follows:

#### 3.2.2 Cropping and Resizing

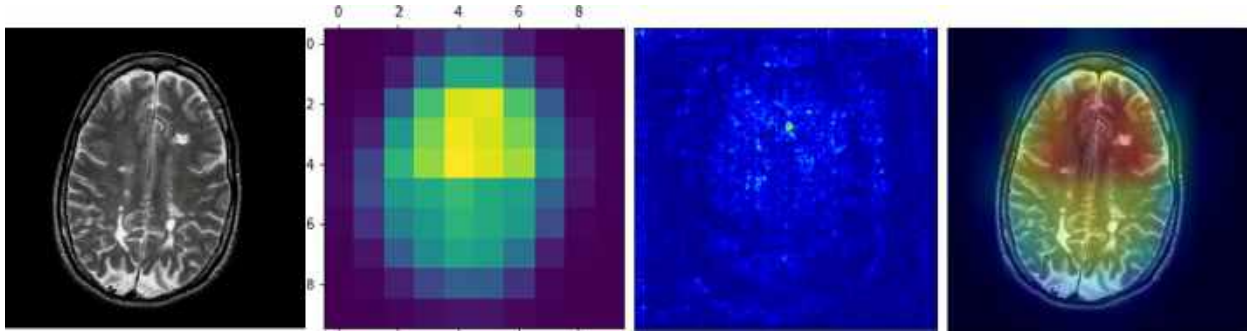
Due to the computational task and limitation of fitting a complete MRI image into the model, we reduce the size of the brain image from ($256 \times 256 \times 3$) to ($131 \times 176 \times 3$). The target size of the brain image is selected in such a way that includes the most of the brain image scanned by the scanner after performing the cropping and resizing effect on the original brain image. In this work, we have worked with the data-driven strategy and performed the cropping on the Brain MRI scans to remove the undesired regions.

– Edge detection: To compute the gradient magnitude of the Brain image, we have applied the Sobel edge detection method to obtain minimum and maximum gradient, inside and at the edge of the various MRI scans of the brain as shown in Fig. 3.

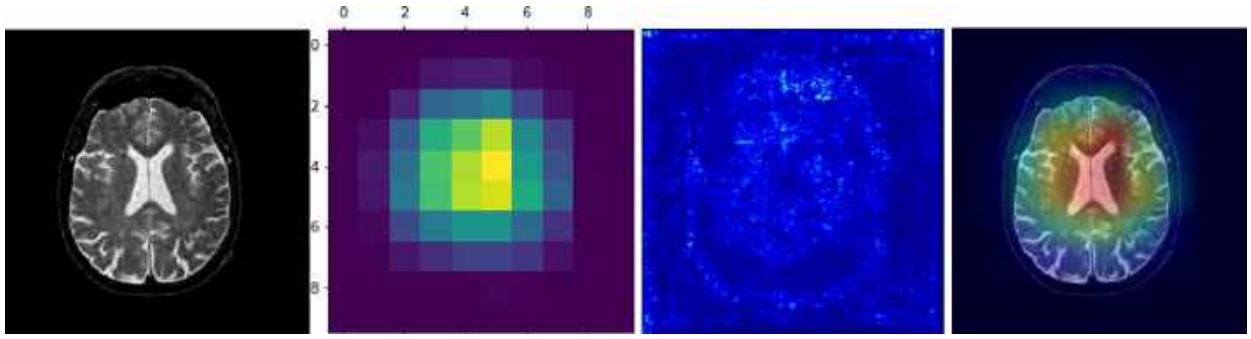
– Thresholding: Once the edge is detected, we perform the thresholding that divides the pixel of the grayscale image and converts the MRI scans into binary images [21].

– Filtering: Further, filtering uses the Sobel filter and removes the noise of low frequencies from the MRI images.

– Segmentation and Contour: Here, we apply a watershed algorithm on the filtered image that extracts the segmented regions by the finding of watershed lines [2].



(a)



(b)

**Fig. 7.** Visualization of multiple sclerosis affected vs. healthy MRI image size (224×224) on the proposed model (a) Class activation map of multiple sclerosis (b) Class activation map for healthy brain MRI

Choosing the Largest Contour of the watershed transformed image and applying grayscale to convert it into binary images [31]. The main steps of the MRI image cropping and resizing procedure are described below.

– Step-1: We have considered the MRI dataset of MS-affected and healthy subjects. Here, the dimension of each input image is (256×256×3).

– Step-2: Further, we have applied the Binary Thresholding on the input image.

– Step-3: Applies the Sobel filter for filtering the thresholding image where the gradient operator are:

$$G_x = [-1,\ 0,\ 1],\quad [-2,\ 0,\ 2],\quad [-1,\ 0,\ 1]$$

$$G_y = [\ 1,\ 2,\ 1],\quad [\ 0,\ 0,\ 0],\quad [-1,\ -2,\ -1]$$

– Step-4: Applies the watershed transformation for segmenting the region in the filtered MRI image. This provides an accurate and effective segmentation of input MRI images.

– Step-5: Applies the chain approximation algorithm on the segmented MRI image and generates green points on all the contours.

– Step-6: Detect each contour find the largest contour on the MRI image and apply crop operation on the largest contour to obtain cropped MRI image of dimension (131×176×3).

### 3.2.3 Image Augmentation

Applying the MobileNet-2D-CNN model to an MRI image of the brain is one of the challenging tasks as a limited amount of training dataset is available. To overcome this problem, data augmentation

**Table 2.** Generated feature map from activation channels

| Activation Layer | No. of Feature Map |
|---|---|
| Mobilenet_1.00_224 | 32 |
| Conv2d | 128 |
| Batch_Normalization_1 | 256 |
| Max_Pooling2d_1 | 512 |
| GlobalAveragePooling2D | 512 |

techniques are used that enhance the size and quality of the training dataset and also improve the performance of the proposed model.

The image augmentation strategy that the network will use for more training data also reduces the overfitting of the proposed model. Image augmentation comes from transformation, color space, random cropping, orientation, mixing images, kernel filters, etc.

The classification accuracy of the deep learning model performs much better in the augmented test data set [27]. We have performed data augmentation on magnetic resonance images, using ImageDataGenerator API from Keras inside the Tensorflow 2.5 before we feed into the MobileNet network.

The augmentation of images includes factors such as rotation, shifting, shear, horizontal flip, vertical flip, and brightness that generate the new training dataset as shown in Fig. 4.

The augmentation operation is performed by random rotation by $45^0$ and then applies to the shift, shearing with the rotated images, and then applying horizontal and vertical flipping of the transformed image. Finally, resize the MRI images that correspond to the input size of the MobileNet model, that is, $(224 \times 224)$ pixels.

### 3.2.4 MobileNet Network Architecture

The Standard convolution model uses a layer stack in which the CNN image features include channel-wise and spatial-wise information. Unlike spatial convolution, depth convolution deals with spatial dimension and depth dimension or the number of channels [14].

A deep separable convolution commonly known as a separable convolution is related to the grouped convolution and inception modules of the Inception family [7]. The depth-wise convolution is followed by a point-wise convolution with (1 $\times$ 1) window and project the new channel space. In MobileNet, the separable convolution allows one to build an image classification model such as MobileNet_224.

In this work, we have used MobileNet as it is a lightweight CNN architecture built primarily from deep separable convolution, as shown in Fig. 5. The small network and low latency achieve good efficiency relative to standard convolution.

The shape of the image has three dimensions of the input feature map F to the convolution layer is (spatial map $\times$ height $\times$ input depth) produces another feature map G that is defined as (spatial map $\times$ height$\times$ output depth) and the size of the convolution kernel K is given as ($D_K \times D_K \times M \times N$) where ($D_K \times D_K$) corresponds to squared spatial dimension of kernel and $M$, and $N$ are input and output depths respectively.

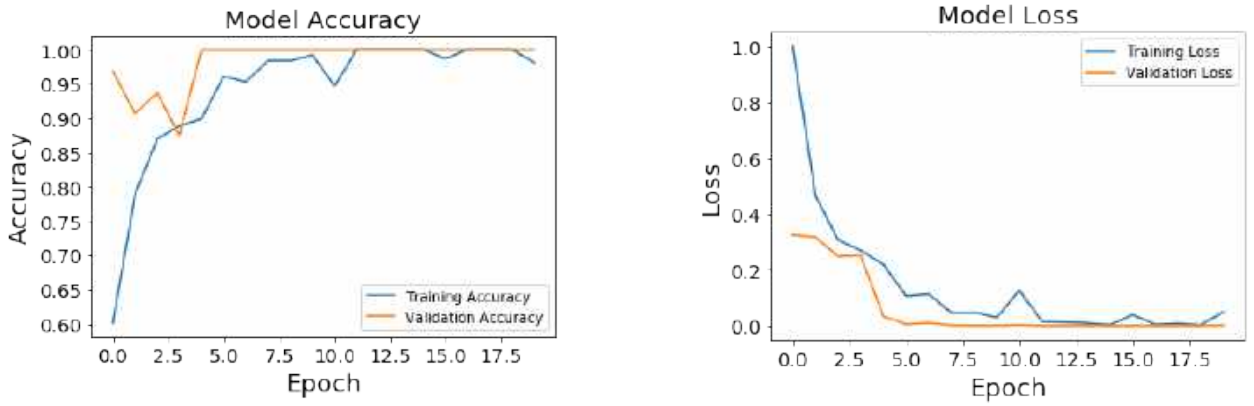
The depth-wise convolutional kernel performs a single convolution on each channel and can be defined as follows:

$$X_{k,\,l,\,m} = \sum_{i,j} K_{i,\,j,\,m} \cdot F_{k+i-1,\,i+j-1,\,m}. \qquad (1)$$

The computational cost of the depth-wise convolution is calculated as:

$$D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F, \qquad (2)$$

where $D_K, D_F, M$ and $N$ are dimensions of the convolutional kernel, spatial width, height, input channel, and output channel, respectively.




(a) (b)

**Fig. 8.** Learning curve of the proposed model (a)Training and validation accuracy (b) Training and validation loss

### 3.2.5 Proposed MobileNet-2D-CNN Architecture

The architecture of the proposed MobileNet 2D CNN is shown in Fig. 5 which is based on a separable filter for depth and performs a single convolution for each input channel. For the target network adding one convolution layer followed by one maximum grouping layer and two fully connected layers of size 256 and 2, the dropout is 0.5% and we have used the root mean square propagation RMSprop optimizer [9] to train the model with the learning rate being 0.0001. Here, RMSprop optimizer is defined as per following:

$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1g_t^2, \tag{3}$$

$$\theta_{t+1} = \theta_t + \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t, \tag{4}$$

where $E[g^2]_t$ is the running average in time $t$ depending on momentum $\gamma$=0.9 [24] and the default value of the learning rate $\eta$ = 0.001 on the previous average $E[g^2]_{t-1}$, gradient of the objective function is denoted by $g_t$ for the update parameter $\theta$ at every time step $t$.

The combination of these 2D-CNN layers is applied along the MobileNet network. The features generated from the MobileNet are further fed to a shallow custom CNN architecture, as shown in Fig. 2. The obtained output shape of the MobileNet MRI image will be input to 2D-CNN.

The proposed MobileNet-2D-CNN model is trained in 20 epochs since we used a training and testing scheme of 80-20%. The proposed model shows an accuracy of 98.15%. Since the data set used in this research is gray images, therefore, the value of pixels in the MRI images is between 0 and 255. The visualization of MRI images shows how perfectly the proposed model extracts the features for the MS classification.

### 3.3 Class Activation Map (CAM)

For a specific class, the class activation map with the global average grouping is applied to the proposed MobileNet-2D-CNN model. These weighted activation maps are generated by the hidden layers of the MRI images.

The network visualization pattern is activated by each unit of the network [16]. CNN learns while being trained to recognize the object [15]. The features obtained are fed into the fully connected layer (FC), regulated by the Softmax activation function, and provide the result of the calculated probabilities for further classification [28].

We have illustrated the activation from the convolutional layer. Consider the activation $k$ of the convolution layer at the location $(i,\ j)$, then the activation function for the MRI image will be denoted as $f_k(i,\ j)$.

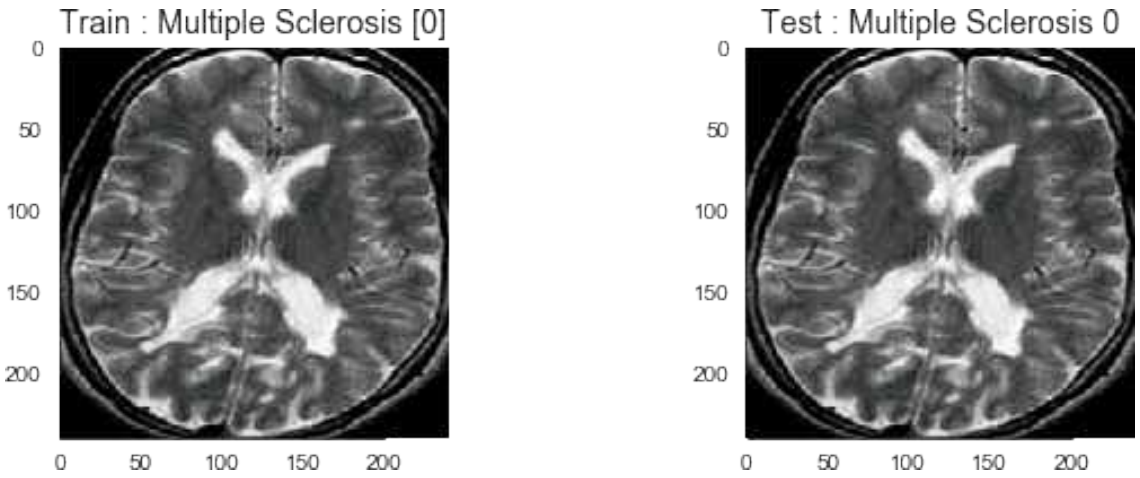



**Fig. 9.** Classified MS affected brain MRI

Here, global average pooling (GAP) at activation $k$ is represented by $G_k$ as shown in Eq. 5:

$$G_k = \sum_{i,j} f_k(i,j). \tag{5}$$

The softmax activation function is used for the computation of class label $C_i$ probablities $P_{C_i}$ [16]:

$$\sigma_{C_i} = \sum_k \omega_k^{C_i} G_k, \tag{6}$$

where, the weight vector $\omega_k^{C_i}$ of the network corresponds to the class label $C_i$ at the activation $k$. Therefore class probablities $P_{C_i}$ and the softmax activation function $\sigma_{C_i}$ is defined as:

$$P_{C_i} = \frac{\exp(\sigma_{C_i})}{\sum_{C_i} \exp(\sigma_{C_i})}, \tag{7}$$

$$\sigma_{C_i} = \sum_k \omega_k^{C_i} \sum_k f_k(i,j). \tag{8}$$

The class activation map (CAM) is obtained by using the weighted feature map governed by the softmax weight $\sigma_{C_i}$ that classifies the heat map corresponding to a specific class.

### 3.3.1 Visualization of Class Activation

In the forward network, the MRI image input $f_k(i,j)$ with height $h$, width $w$, and depth $d$ is processed through the proposed MobileNet–2D-CNN model. The MRI image tensor $(h \times w)$ maps each pixel dimension to its corresponding color class $C_i$. Additionally, the classifier maps the input MRI image to the class saliency map $H \in \mathbb{R}^{h \times w}$ and applies the activation function $G_k$ to each pixel of the input image $f_k(i,j)$. The weighted neurons are computed as the gradient of the softmax activation function $\sigma_{C_i}$ with respect to the activation of the feature map $f_k$ from the convolutional layer, as defined below:

$$\omega_k^{C_i} = \left.\frac{\partial \sigma_{C_i}}{\partial f_k}\right|_{f_k(i,j)}. \tag{9}$$

Taking the partial derivative of $G_k$ with respect to $f_k(i,j)$, that is:

$$\frac{1}{N} = \frac{\partial G_k}{\partial f_k(i,j)}, \tag{10}$$

$$H = \text{relu}\left\{\sum_k \omega_k^{C_i} f_k(i,j)\right\}. \tag{11}$$

This formulation of the computation allows us to generate the visualization of the saliency map of the multiple sclerosis (MS) MRI image.

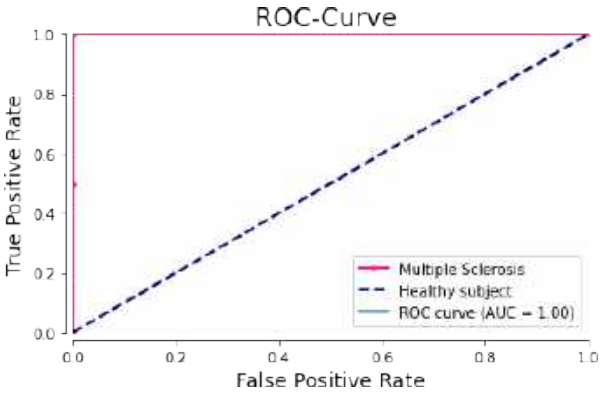



**Fig. 10.** ROC curve estimates the AUC between 0 - 1 with the binary classifier and plots the true positive rate vs. false positive rate

## 4 Experiments and Results

The proposed model evaluates the classification of Multiple Sclerosis present in MRI images. Based on the formulae described above, the proposed model considers the size of the MRI images (224×224 ×3) that includes 475 training images for the training set, 19 images for validation to perform the support dataset and 95 images for the test dataset. We have observed the hidden layer gradient images, the MS classification, and the performance of the proposed model. We begin by presenting the class activation map using gradients to interpret the proposed network, experimental training strategy, and classification followed by model evaluation.

### 4.1 Implementation Details

The experimental setup of the MobileNet-2D-CNN model has been designed and trained on MRI images as shown in Fig. 2. The MobileNet input layer has a size of (224×224×3) to preprocess the MRI image data. The next layer is the 2D convolutional layer with kernel size (3×3) it is determined the receptive field that received input with the dimension of (7×7×1024) then applying the ReLU activation function to the output of the convolution, after that the batch normalization layer takes the input size of the data (5×5×32) it is used to normalize the data and reduces the data loss between the processing layers, a max pooling layer is used to downsample to the output of size (1×1×32). The global average pooling layer converts the 3D feature map matrix from the maximum pooling layer to the vector 1D and generates a single feature map.

Subsequently, the two fully connected (FC) layers are used for classification purposes. Here, the first FC layer has 256 output neurons followed by the dropout layer with probability = 0.5, and the next FC layer consists of 256 input neurons with 2 output neurons for the two-class classification.

We have trained the proposed MobileNet–2D-CNN model with the binary cross-entropy loss function and the RMSprop optimizer, where the batch size is 32, and 20 epochs are applied to the MRI image data. To classify MS, the proposed MobileNet 2D-CNN architecture is implemented by using Tensorflow and Keras open source library.

The proposed model is trained on the powerful NVIDIA Geforce GTX 1080 GPU. All parameters used in the proposed network are summarized in Table 1. The classification accuracy obtained from the proposed MobileNet 2D-CNN model is 98.15%. Here, Fig. 8 shows the training and validation accuracy of the proposed model.

### 4.2 Grad-CAM Visualization

The proposed MobileNet–2DCNN architecture is trained to produce heat maps from input MRI images. These heat maps detect the location of the MS lesion on the MRI image as per Fig. 7. The heat maps generated from the convolutional layer use a single input channel due to grayscale MRI images. Here, the convolutional layer considers the weighted average output.

In each channel, the feature map uses the weight in the FC layer. The softmax layer performs the computation for the class prediction of the heatmaps. The output of a softmax layer gradient with respect to each channel forms a feature map of a specific layer that displays a gradient of the respective output channel. We have obtained feature maps generated from each activated layer as per Table 2.

These gradient feature maps of MRI images are fed into the Global Average Pooling (GAP) layer that considers the size of the MRI image tensor as ($5\times5\times32$) and performs the averaging across the ($5\times5$) convolution. The average value of each of the input channels generates one output channel, i.e. a one-dimensional tensor with 32 images. These weighted feature maps 2D, as generated by the GAP layer, are used as heat maps.

### 4.3 Classification of MS Lesion

MRI images are classified using the proposed model, which predicts class labels as illustrated in Fig. 9. The model is trained and tested on a dataset of 475 preprocessed MRI images, including both healthy controls and MS-affected individuals. This dataset is split into 80% for training and 20% for testing, with 5% of the training data reserved for validation. The classification utilizes a 2D CNN to identify MS lesions, as demonstrated in Fig. 9.

### 4.4 Model Evaluation

In the binary classification, the prediction of classes for the computed probabilities is based on the continuous variable. Probability is classified as positive if the computed probabilities $>$ threshold value otherwise negative. The classification performance of the proposed model is measured using the area under the curve (AUC).

The receiver operating characteristic (ROC) curve between the TPR and FPR, computes the area under the ROC curve [6, 19]. The AUC ranges from 0.5 to 1.0, generally interpreted as the probability that the diseased subject is randomly selected which has a higher test value compared to the random selection of the healthy subject [13].

The AUC value measures the overall performance of the classification model. Here, Fig. 9 shows that the classifier of the proposed MobileNet-2D CNN has AUC = 1.00, which shows that the performance of the model is based on the MRI scan data and is perfectly classified by the proposed model. A model with multiple sclerosis is shown by an orange curve that travels from the bottom left to the top right and above the threshold diagonal line.

## 5 Conclusion and Future Work

We have proposed a MobileNet - 2D-CNN model for the detection and classification of MS-affected lesions in brain MRI images. The proposed model uses the threshold-based extraction method from the MRI image. Further, we have applied the data augmentation methods for generating the dataset during the pre-processing steps.

The learning process of the proposed model depends mainly on the selection of the training data set. Using the augmented data to experiment with our proposed MobileNet–2D-CNN architecture as shown in Fig. 6 generates the gradient class activation map for the detection and classification of MS lesions from MRI images.

The experimental results obtained to find the MS-affected classification from the test dataset are shown in Fig. 8. In these results, we have achieved a classification accuracy of 98.15% from the binary classifier. An ROC curve shows the classification performance with AUC = 1.00.

For both detection and classification, it is expected that an in-depth study can achieve a similar quality of feature that can be achieved by a nonlinear deep learning model. For that reason, future research towards creating a more refined structure that can take the findings forward to the systematic exploration of the lesion map in terms of clinical application.

## References

1. **Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D. L., Erickson, B. J. (2017).** Deep learning for brain MRI segmentation: State of the art and future directions. Journal of Digital Imaging, Vol. 30, No. 4, pp. 449–459. DOI: 10.1007/s10278-017-9983-4.
2. **Beare, R., Chen, J., Adamson, C. L., Silk, T., Thompson, D. K., Yang, J. Y. M., Anderson, V. A., Seal, M. L., Wood, A. G. (2013).** Brain extraction using the watershed transform from markers. Frontiers in Neuroinformatics, Vol. 7. DOI: 10.3389/fninf.2013.00032.

3. **Benedict, R. H. B., Amato, M. P., DeLuca, J., Geurts, J. J. G. (2020).** Cognitive impairment in multiple sclerosis: Clinical management, MRI, and therapeutic avenues. The Lancet Neurology, Vol. 19, No. 10, pp. 860–871. DOI: 10.1016/s1474-4422(20)30277-5.
4. **Birenbaum, A., Greenspan, H. (2017).** Multi-view longitudinal CNN for multiple sclerosis lesion segmentation. Engineering Applications of Artificial Intelligence, Vol. 65, pp. 111–118. DOI: 10.1016/j.engappai.2017.06.006.
5. **Chhatbar, P. Y., Kara, P. (2013).** Improved blood velocity measurements with a hybrid image filtering and iterative Radon transform algorithm. Frontiers in Neuroscience, Vol. 7, pp. 106. DOI: 10.3389/fnins.2013.00106.
6. **Cho, H., Matthews, G., Harel, O. (2018).** Confidence intervals for the area under the receiver operating characteristic curve in the presence of ignorable missing data. International Statistical Review, Vol. 87, No. 1, pp. 152–177. DOI: 10.1111/insr.12277.
7. **Chollet, F. (2017).** Xception: Deep learning with depthwise separable convolutions. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1800–1807. DOI: 10.1109/cvpr.2017.195.
8. **Ding, Y., Acosta-Sánchez, R., Enguix-Chiral, V., Suffren, S., Ortmann, J., Luck, D., Dolz, J., Lodygensky, G. (2020).** Using deep convolutional neural networks for neonatal brain image segmentation. Frontiers in Neuroscience, Vol. 14, pp. 207. DOI: 10.3389/fnins.2020.00207.
9. **Duchi, J., Hazan, E., Singer, Y. (2011).** Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, Vol. 12, No. 61, pp. 2121–2159.
10. **Gupta, P. K., Ören, T., Singh, M. (2018).** Predictive intelligence using big data and the internet of things. IGI Global.
11. **Gupta, P. K., Tyagi, V., Singh, S. K. (2017).** Predictive computing and information security. Springer Singapore. DOI: 10.1007/978-981-10-5107-4.
12. **Guttmann, C. R., Ahn, S. S., Hsu, L., Kikinis, R., Jolesz, F. A. (1995).** The evolution of multiple sclerosis lesions on serial MR. American Journal of Neuroradiology, Vol. 16, No. 7, pp. 1481–1491.
13. **Hanley, J. A., McNeil, B. J. (1982).** The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology, Vol. 143, No. 1, pp. 29–36. DOI: 10.1148/radiology.143.1.7063747.
14. **Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H. (2017).** MobileNets: Efficient convolutional neural networks for mobile vision applications. DOI: 10.48550/arXiv.1704.04861.
15. **Hoyer, L., Munoz, M., Katiyar, P., Khoreva, A., Fischer, V. (2019).** Grid saliency for context explanations of semantic segmentation. Proceedings of the 33rd International Conference on Neural Information Processing Systems, pp. 6462–6473.
16. **Kwaśniewska, A., Rumiński, J., Rad, P. (2017).** Deep features class activation map for thermal face detection and tracking. Proceedings of the 10th International Conference on Human System Interactions, pp. 41–47. DOI: 10.1109/HSI.2017.8004993.
17. **Lecun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998).** Gradient-based learning applied to document recognition. Proceedings of the IEEE, Vol. 86, No. 11, pp. 2278–2324. DOI: 10.1109/5.726791.
18. **Lu, S. Y., Wang, S. H., Zhang, Y. D. (2020).** A classification method for brain MRI via mobilenet and feedforward network with random weights. Pattern Recognition Letters, Vol. 140, pp. 252–260. DOI: 10.1016/j.patrec.2020.10.017.
19. **López-Cabrera, J., Lorenzo-Ginori, J. (2017).** Automatic classification of traced neurons using morphological features.

Computacion y Sistemas, Vol. 21, No. 3, pp. 537–544. DOI: 10.13053/CyS-21-3-2495.

20. **McKinley, R., Wepfer, R., Grunder, L., Aschwanden, F., Fischer, T., Friedli, C., Muri, R., Rummel, C., Verma, R., Weisstanner, C., Wiestler, B., Berger, C., Eichinger, P., Muhlau, M., Reyes, M., Salmen, A., Chan, A., Wiest, R., Wagner, F. (2020).** Automatic detection of lesion load change in multiple sclerosis using convolutional neural networks with segmentation confidence. NeuroImage: Clinical, Vol. 25, pp. 102104. DOI: 10.1016/j.nicl.2019.102104.
21. **Oliva, D., Cuevas, E., Pajares, G., Zaldivar, D., Osuna, V. (2014).** A multilevel thresholding algorithm using electromagnetism optimization. Neurocomputing, Vol. 139, pp. 357–381. DOI: 10.1016/j.neucom.2014.02.020.
22. **Pasa, F., Golkov, V., Pfeiffer, F., Cremers, D., Pfeiffer, D. (2019).** Efficient deep network architectures for fast chest x-ray tuberculosis screening and visualization. Scientific Reports, Vol. 9, No. 1. DOI: 10.1038/s41598-019-42557-4.
23. **Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Takemura, H., Petrilli-Barcelo, A. E. (2021).** High-resolution reconstructions of aerial images based on deep learning. Computación y Sistemas, Vol. 25, No. 4, pp. 739–749. DOI: 10.13053/cys-25-4-4047.
24. **Ruder, S. (2016).** An overview of gradient descent optimization algorithms. DOI: 10.48550/ARXIV.1609.04747.
25. **Saurabh, S., Gupta, P. K. (2022).** Non-linear behavior of CNN model interpretation using saliency map. Proceedings of the 7th International Conference on Parallel, Distributed and Grid Computing, pp. 733–738. DOI: 10.1109/PDGC56933.2022.10053135.
26. **Saurabh, S., Gupta, P. K. (2023).** Deep learning-based modified bidirectional LSTM network for classification of ADHD disorder. Arabian Journal for Science and Engineering, Vol. 49, No. 3, pp. 3009–3026. DOI: 10.1007/s13369-023-07786-w.
27. **Shorten, C., Khoshgoftaar, T. M. (2019).** A survey on image data augmentation for deep learning. Journal of Big Data, Vol. 6, No. 1, pp. 1–48. DOI: 10.1186/s40537-019-0197-0.
28. **Simonyan, K., Vedaldi, A., Zisserman, A. (2014).** Deep inside convolutional networks: Visualising image classification models and saliency maps. Workshop at International Conference on Learning Representations.
29. **Sinha, D., El-Sharkawy, M. (2019).** Thin MobileNet: An enhanced mobilenet architecture. Proceedings of the IEEE 10th Annual Ubiquitous Computing, Electronics Mobile Communication Conference, pp. 280–285. DOI: 10.1109/UEMCON47517.2019.8993089.
30. **Tousignant, A., Lemaître, P., Precup, D., Arnold, D. L., Arbel, T. (2019).** Prediction of disease progression in multiple sclerosis patients using deep learning analysis of MRI data. Proceedings of the 2nd International Conference on Medical Imaging with Deep Learning, Vol. 102, pp. 483–492.
31. **Varela, E., Moya-Sanchez, E. U., Aguilar-Meléndez, A., Castillo Reyes, O., Vázquez-Santacruz, E., Salazar-Colores, S., Cortés, U. (2019).** Detection, counting, and classification of visual ganglia columns of drosophila pupae. Computación y Sistemas, Vol. 23, No. 2, pp. 391–397. DOI: 10.13053/CyS-23-2-3200.
32. **Zhang, Y. D., Pan, C., Sun, J., Tang, C. (2018).** Multiple sclerosis identification by convolutional neural network with dropout and parametric ReLU. Journal of Computational Science, Vol. 28, pp. 1–10. DOI: 10.1016/j.jocs.2018.07.003.

*Article received on 25/03/2022; accepted on 21/07/2024.*
**Corresponding author is Sudhanshu Saurabh.*

ISSN 2007-9737

# Natural Language Processing Approach Using a Neural Network Ensemble (CNN-HSNN) for Skin Cancer and Multi-Disease Classification

Joshua R. G. Guerrero-Rangel[1], Grigori Sidorov[1], Christian E. Maldonado-Sifuentes[2,*], Mariano Vargas-Santiago[2], M. Cristina Ortega-García[3], Diana A. Leon-Velasco[2]

[1] Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

[2] Consejo Nacional de Humanidades, Ciencia y Tecnología,
Mexico

[3] Transdisciplinary Research for Augmented Innovation Laboratory,
Mexico

[4] Instituto Tecnológico y de Estudios Superiores de Monterrey,
Deparmento de Ciencias,
Mexico

christian.maldonado, mariano.vargas}@conahcyt.mx, jguerrero@gmail.com, sidorov@cic.ipn.mx, cristina.ortega@trai-l.com, assaely.leon@tec.mx

**Abstract.** Dermatological diseases, including skin cancer, represent a significant challenge for global health systems. Early and accurate diagnosis is crucial to improve patient outcomes and reduce treatment costs. This study leverages an ensemble system combining Convolutional Neural Network (CNN) and Hybrid Sequential Neural Network (HSNN) models to accurately classify various dermatological diseases, including skin cancer, Dermatitis Atopica, Melasma (Cloasma), and Vitiligo. The CNN model processes skin cancer data, while the HSNN model handles the other diseases using a combination of embedding, LSTM, and dense layers. The ensemble system achieved a global F1-score of 95.45%, demonstrating balanced diagnostic precision across all diseases. Precision, recall, and F1-scores were consistently high across the different diseases, underscoring the ensemble system's robustness. These results provide a reliable decision-support tool for early diagnosis and personalized treatment of dermatological diseases, ultimately contributing to improved patient outcomes and optimized healthcare efficiency. Future work aims to expand the framework to cover additional dermatological conditions and integrate both text and image data for comprehensive diagnostic analysis.

**Keywords.** Machine learning, NLP, cancer, skin affections, DNN ensemble.

## 1 Introduction

Dermatological diseases, such as skin cancer, Dermatitis Atopica, Melasma, and Vitiligo, have emerged as significant global health concerns due to their rising prevalence and the challenges associated with accurate diagnosis and treatment. According to the World Health Organization (WHO), skin cancer remains an emergent issue,

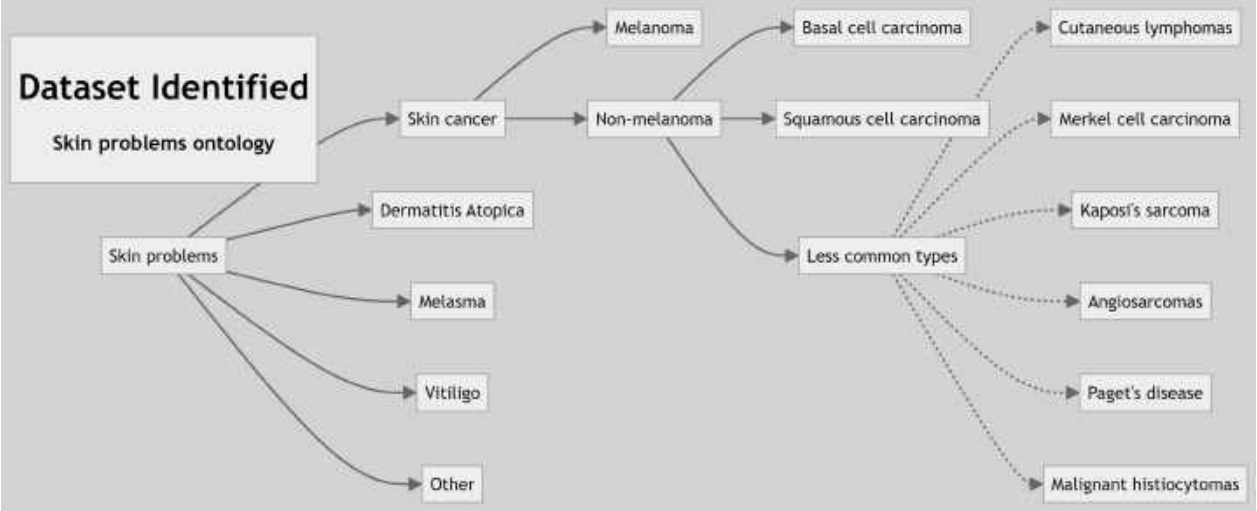



**Fig. 1.** Ontology of skin problems found in our dataset, prepared by the authors

representing a considerable proportion of all cancer cases, particularly in the United States, Australia, and Canada [24, 4].

The rising incidence of skin cancer has led to substantial annual mortality, particularly for melanoma, which is projected to increase further in the coming years [24].

Environmental factors such as ozone layer depletion, increased ultraviolet (UV) radiation, lifestyle habits like smoking, and infections significantly contribute to skin cancer risk. Despite being less common, melanoma remains one of the deadliest skin cancers due to its high potential for metastasis.

Similarly, non-cancerous dermatological conditions like Dermatitis Atopica, Melasma, and Vitiligo also significantly impact patients' quality of life and healthcare systems globally.

The diagnosis and classification of these diseases often rely on clinical assessment, which can be subjective and lead to inconsistent results.

The increasing demand for more accurate and objective diagnostic methods has driven the adoption of advanced computer-assisted techniques, leveraging Natural Language Processing (NLP) and Machine Learning (ML).

### 1.1 Contributions of this Study

1. **Ensemble Methodology**: We present a novel ensemble system that integrates a Convolutional Neural Network (CNN) and a Hybrid Sequential Neural Network (HSNN) to enhance the accuracy of dermatological disease classification. This system combines the strengths of both models to process textual data and sophisticated classification.
2. **Textual Data Analysis**: Our study leverages advanced NLP techniques like Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) to represent clinical text data, enabling our ensemble model to accurately understand patterns in medical notes.
3. **Diagnostic Precision**: This methodology significantly improves diagnostic precision across skin cancer, Dermatitis Atopica, Melasma, and Vitiligo, achieving an overall F1-score of 95.45%.

   The ensemble system provides reliable decision support to healthcare professionals, reducing diagnosis errors and wait times.
4. **Implications for Smart Healthcare**: Our research contributes to the broader

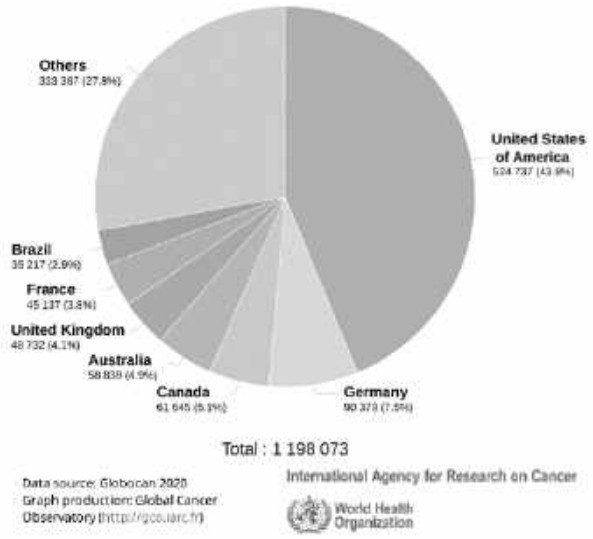



**Fig. 2.** Global estimated number of new cases in 2020, non-melanoma skin cancer, both sexes, all ages

development of smart healthcare systems through the sophisticated classification of various dermatological conditions. By offering reliable, early-stage diagnoses, we enhance treatment outcomes and support healthcare professionals in efficiently combating these diseases.

### 1.2 Paper Structure

The remainder of this paper is structured as follows:

- Section 2 discusses relevant literature related to dermatological disease classification and diagnosis.
- Section 3 details our method and methodology.
- Section 4 presents our findings from traditional statistical and machine learning-based analyses.
- Finally, Section 5 provides conclusions and future work.

## 2 State of the Art

In this section, we present the current state of the art in tasks related to cancer detection, with a particular emphasis on text-based methods for machine learning detection of skin cancer, complemented by recent advancements in deep learning models for more precise classification.

The application of Natural Language Processing (NLP) and Machine Learning (ML) techniques has shown promising advancements in the classification of various types of cancer. Recent research has increasingly focused on leveraging these techniques for skin cancer classification, utilizing both textual data and imaging.

### 2.1 Advancements in Deep Learning for Skin Cancer Detection

Recent studies have made significant strides in improving the accuracy of skin cancer detection through the integration of deep learning models with existing datasets and novel preprocessing techniques.

#### 2.1.1 Enhanced Convolutional Models

Zia et al. [31] improved the diagnosis of skin cancer by integrating extra convolution layers into two pre-trained deep learning models, MobileNetV2 and DenseNet201. The modified DenseNet201 model achieved 95.50% accuracy in identifying benign and malignant skin lesions, outperforming the MobileNetV2 model, which reached 91.86% accuracy.

These models were trained using an updated dataset from the ISIC repository, with augmentation techniques such as the introduction of Gaussian noise to enhance the dataset's variance. Gouda et al. [9] utilized DCNN models to detect primary tumors, achieving accuracies up to 85.8% with InceptionV3. The ISIC 2018 dataset served as the basis for model training, with image enhancement techniques employed during the preprocessing stage to improve model performance. Dorj et al. [6] proposed a combination of a pre-trained AlexNet CNN model with an ECOC SVM classifier, achieving

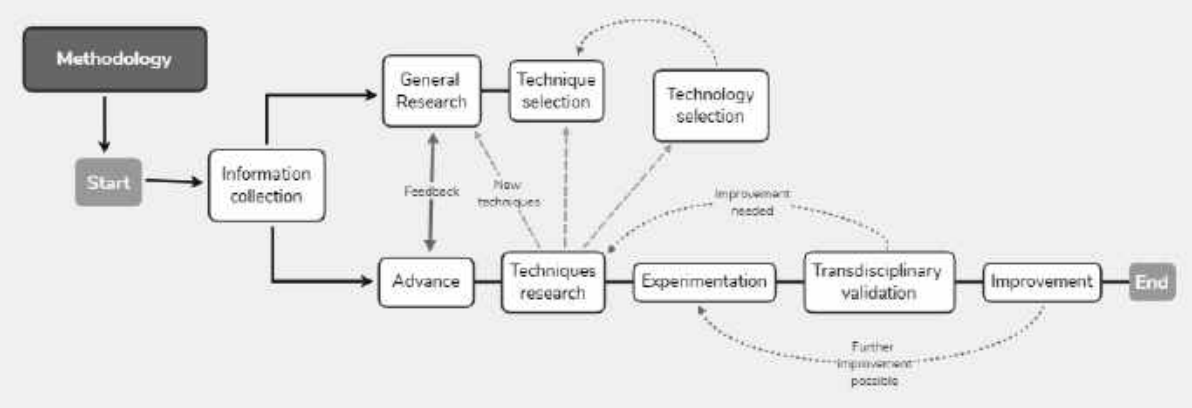



**Fig. 3.** General view of our methodology

a maximum accuracy of 95.1%. This study highlighted the potential of combining deep learning models with classical machine learning classifiers for enhanced skin cancer detection.

### 2.1.2 CNN-based Models for Classification

Fu'adah et al. [7] and Senthil Kumar et al. [16] both employed CNN-based models to detect skin cancer, achieving accuracies of 83% and 88%, respectively. These studies underline the effectiveness of CNNs in differentiating between malignant and benign skin lesions.

Stieler et al. [25] and Wei et al. [29] explored the application of DNNs and lightweight models for skin cancer detection, focusing on domain-specific explanations and high-precision lesion segmentation. Although some studies did not report specific accuracy metrics, the qualitative results demonstrate the models' robustness in skin lesion segmentation.

### 2.1.3 Novel Approaches and Data Augmentation Techniques

Ameri et al. [3], Nawaz et al. [19], and Abayomi-Alli et al. [1] presented innovative approaches to skin cancer diagnosis using deep learning, including the use of transfer learning and novel data augmentation techniques. These studies achieved significant improvements in accuracy, sensitivity, and specificity, highlighting the potential of deep learning in advancing skin cancer detection and classification.

## 2.2 Atopic Dermatitis

Atopic dermatitis (AD) is a chronic inflammatory skin condition affecting millions worldwide. Accurate detection, diagnosis, and classification of this condition are crucial for effective management and treatment. Recent advancements in machine learning, particularly convolutional neural networks (CNNs) and hybrid sequential neural networks (HSNNs), have significantly improved the accuracy and efficiency of AD diagnosis.

Gautam et al. [8] present a skin disease detection system using a hybrid convolutional neural network capable of diagnosing early-stage skin diseases. By combining convolutional layers and fully connected layers, their deep neural network accurately identifies early signs of various skin diseases.

Shivasharan evaluates a hybrid deep learning system for predicting dermatological conditions using convolutional neural networks and unsupervised learning techniques. The study employs a combination of CNNs and clustering algorithms to identify dermatological conditions effectively.

Saifan and Jubair [22] develop a sequential convolutional neural network model for accurate classification of six skin diseases, including dermatitis. Their sequential CNN model employs

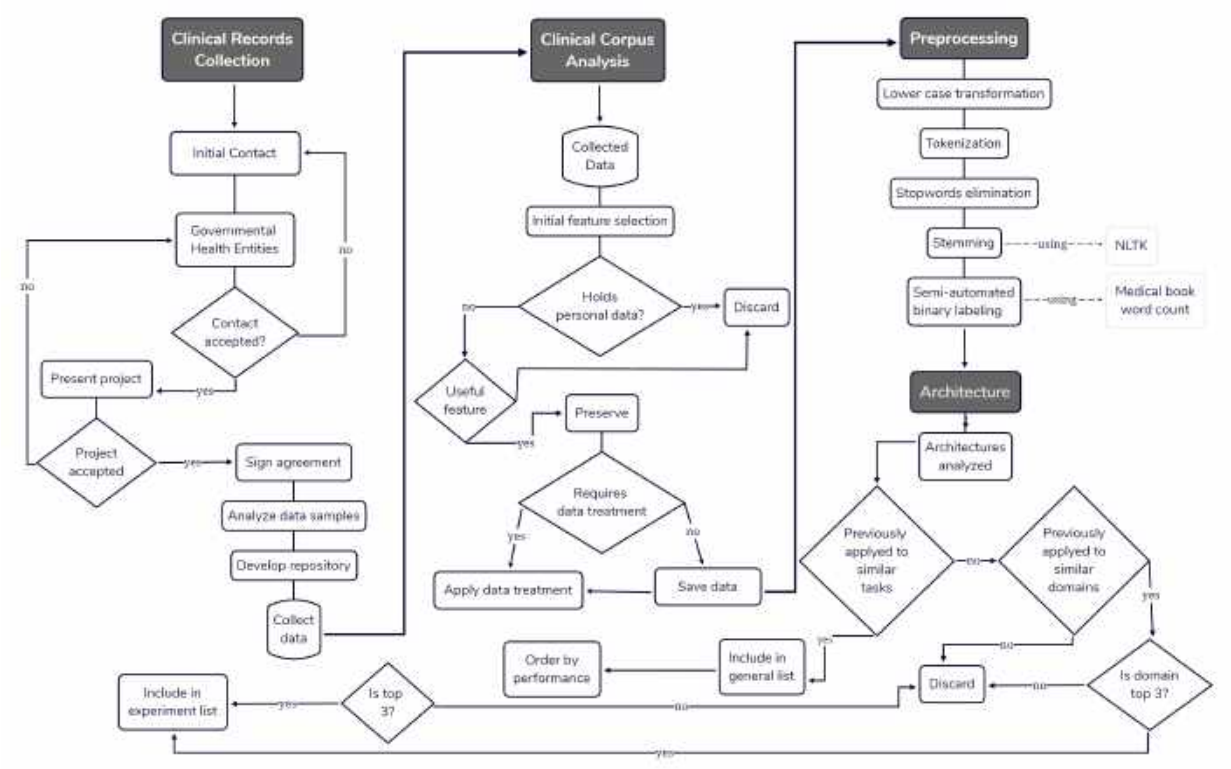



**Fig. 4.** General view of our transdiciplinary method

multiple convolutional layers followed by dense layers to achieve accurate classification.

Hammad et al. [13] propose an efficient sequential model based on attention-enhanced convolutional neural networks for detecting eczema and psoriasis. The attention mechanism improves the model's ability to focus on relevant regions for accurate disease classification.

Chan et al. [5] provide an overview of machine learning applications in dermatology, highlighting the use of convolutional neural networks and reinforcement learning for diagnosis. The study reviews recent advances in machine learning models for dermatology, with a focus on CNNs and reinforcement learning. Kshirsagar et al. [15] suggest a hybrid convolutional neural network approach for the automated diagnosis and prognosis of skin diseases. Their hybrid approach combines convolutional neural networks with decision trees for improved diagnosis.

Guimarães et al. [10] utilize a convolutional neural network-based approach combined with multiphoton tomography for the accurate diagnosis of atopic dermatitis. The combination of multiphoton tomography and CNNs provides high-resolution images for precise diagnosis.

Li et al. [17] introduce a hybrid deep model combining particle swarm optimization with fuzzy k-nearest neighbor for predicting atopic dermatitis. Their study leverages particle swarm optimization to enhance fuzzy k-nearest neighbor classifiers.

Gunwant et al. [11] implement EfficientNet-B0 convolutional neural network for precise skin disease classification. EfficientNet-B0 is used to efficiently classify various skin diseases, achieving high accuracy.

Pangti et al. [20] combine a focal loss function with a hybrid convolutional neural network to build a decision support system for dermatological disease diagnosis. The system uses a custom

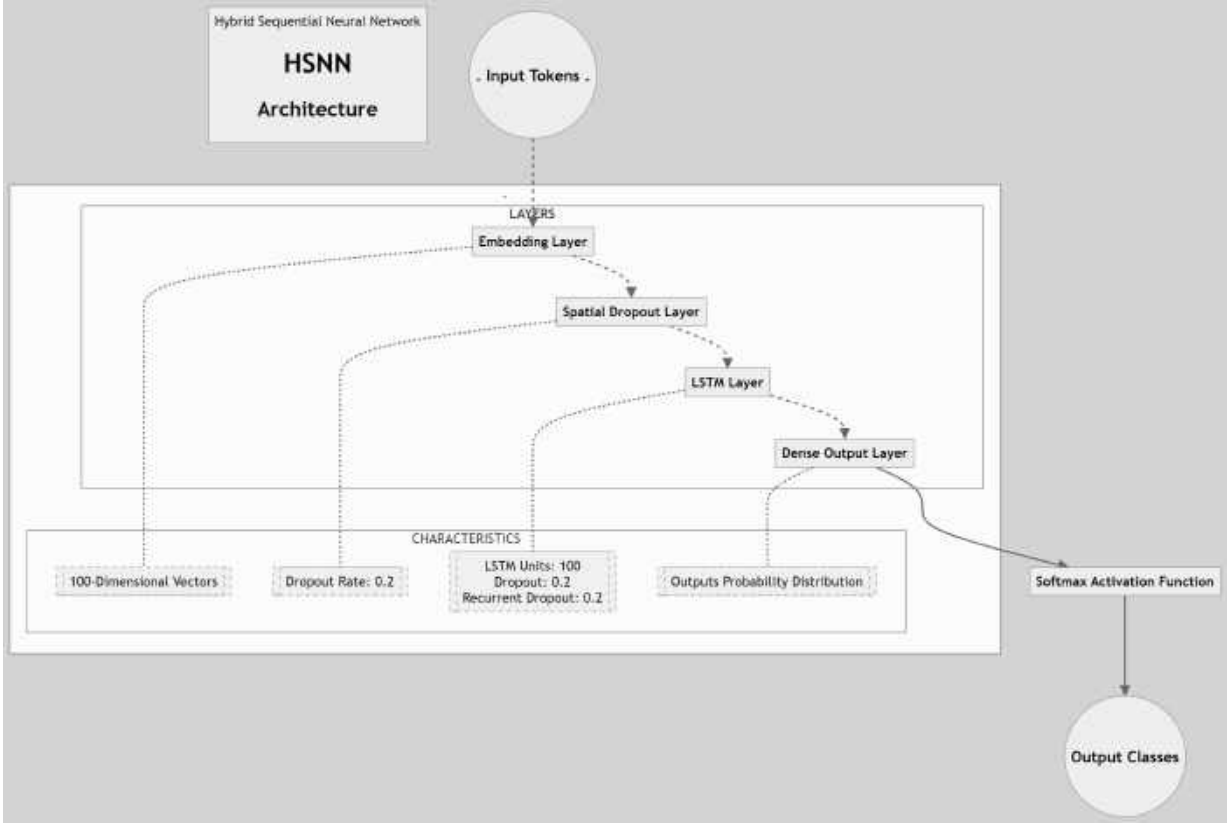



**Fig. 5.** Diagram of our HSNN architecture

loss function to improve diagnostic accuracy in dermatology.

Rasheed et al. [21] employ a hybrid deep neural network combining convolutional neural networks and histogram probabilities to classify eczema. The combination of CNNs and histogram probabilities improves the classification of eczema.

Thomsen et al. [27] provide a systematic review focusing on machine learning models for dermatological diagnosis, including convolutional neural networks, sequential models, and hybrid sequential neural networks. Their review highlights the state-of-the-art machine learning models for dermatological disease diagnosis and prognosis.

## 2.3 Melasma

Melasma (also known as cloasma) is a common pigmentary disorder that causes brown or gray-brown patches on the skin, particularly on the face. Accurate detection, diagnosis, and classification are crucial for effective management and treatment. Recent advancements in machine learning, particularly convolutional neural networks (CNNs) and hybrid sequential neural networks (HSNNs), have shown potential in improving the diagnosis of melasma.

### 2.3.1 Non-tumorous Facial Pigmentation Classification

Tian et al. [28] presented a multi-view convolutional neural network with an attention mechanism to classify non-tumorous facial pigmentation conditions, including melasma.

The model leverages multiple views of the affected areas to improve diagnostic accuracy.

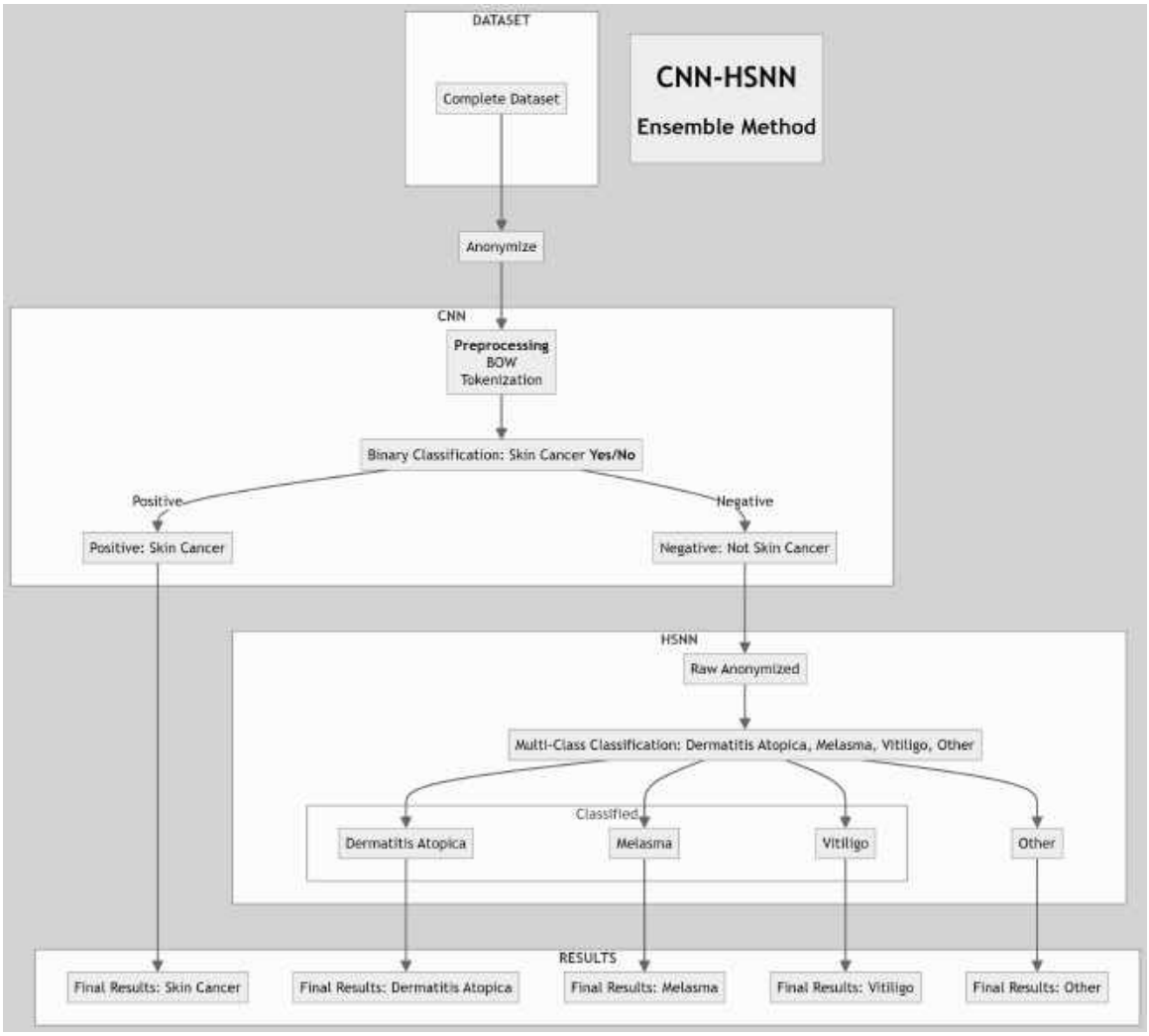



**Fig. 6.** Technical view of our ensemble method

### 2.3.2 Prediction of Susceptibility to Facial Post-inflammatory Hyperpigmentation

Sun et al. [26] developed a neural network model to predict susceptibility to facial post-inflammatory hyperpigmentation, including conditions such as melasma, chloasma, solar freckle-like nevus, and others. The model uses clinical data to identify high-risk patients.

### 2.3.3 Clinical Context in Cosmetic Dermatologic Surgery

Alexiades and Zubek [2] provide a comprehensive overview of cosmetic dermatologic surgery techniques, including the diagnosis of melasma. Although not directly focused on machine learning, the book provides clinical context for understanding the importance of accurate detection and diagnosis of melasma.

These studies illustrate a diverse range of applications for machine learning and neural networks in melasma diagnosis.

**Table 1.** Performance of the ensemble system

| Disease | Precision | Recall | F1 Score |
|---|---|---|---|
| Skin Cancer | 90.2% | 93.1% | 91.1% |
| Dermatitis Atopica | 94.68% | 98.35% | 96.48% |
| Melasma (Cloasma) | 94.79% | 95.92% | 95.35% |
| Vitiligo | 98.51% | 88.50% | 93.23% |

From attention-based convolutional neural networks to predictive models for post-inflammatory hyperpigmentation, the field is continually advancing to improve diagnostic accuracy.

### 2.4 Vitiligo

Vitiligo is a chronic skin disorder characterized by the loss of pigmentation, leading to white patches on the skin. Accurate detection, diagnosis, and classification are crucial for effective management and treatment. Recent advancements in machine learning, particularly convolutional neural networks (CNNs) and hybrid sequential neural networks (HSNNs), have shown great potential in enhancing vitiligo diagnosis.

Shivasharan [23] evaluates a hybrid deep learning system that employs convolutional neural networks (CNNs) and unsupervised learning techniques for predicting various dermatological conditions, including vitiligo. The study demonstrates the potential of CNNs for accurate diagnosis.

Saifan and Jubair [22] developed a sequential convolutional neural network model for accurate classification of six skin diseases, including vitiligo. The sequential CNN model employs multiple convolutional layers followed by dense layers to achieve accurate classification.

Kantoria et al. [14] propose a deep learning-based classification model for vitiligo diagnosis. They demonstrate the effectiveness of a hybrid CNN model in identifying vitiligo-infected lesions.

Guo et al. [12] propose a hybrid deep learning model to detect and assess vitiligo lesions' severity. The model combines three deep convolutional neural networks (DCNNs) to improve diagnosis.

Zhang et al. [30] provide a comprehensive review of machine learning applications for vitiligo diagnosis. They explore how hybrid artificial intelligence models, including residual deep convolutional neural networks (RDCNN), enhance diagnostic accuracy.

Li et al. [18] provide a detailed overview of computer-based algorithms used for the detection of vitiligo. They discuss hybrids and enhanced augmented autonomous intelligence models, highlighting the importance of CNNs and transformers in dermatology image analysis.

Pangti et al. [20] develop a convolutional neural network-based mobile phone application for diagnosing common dermatological diseases, including vitiligo. The hybrid CNN model incorporates a custom loss function to enhance diagnostic accuracy.

These studies highlight the application of machine learning and neural networks in vitiligo diagnosis. By employing CNN-based architectures, including hybrid and attention-enhanced models, researchers have made significant strides in improving diagnostic accuracy and lesion severity assessment.

## 3 Proposed Method

### 3.1 Methodology

Our research methodology follows a rigorous, iterative process to select, refine, and validate the best techniques for diagnosing dermatological diseases, including skin cancer, with a focus on using Natural Language Processing (NLP) and machine learning techniques. The methodology is structured as follows:

1. **Information Collection**: The research begins with collecting relevant data from various sources. This includes clinical records, medical documentation, and expert input.

2. **General Research and Advance**: We conduct comprehensive literature reviews to understand the current state of the art. These insights help guide our research direction and inform the selection of appropriate techniques.
3. **Techniques Research**: Through further analysis, we identify potential algorithms, models, and architectures that are promising for our dataset. This step involves exploring a diverse set of models and refining them iteratively.
4. **Technique and Technology Selection**: From the researched techniques, we select those most suitable for further experimentation. Feedback loops with general research inform adjustments, while new techniques are integrated into the analysis.
5. **Experimentation**: We perform systematic experimentation, testing different models and parameter configurations. The results are continuously assessed to identify areas where improvements can be made.
6. **Transdisciplinary Validation**: Experts from different domains validate the results, providing critical input for refining the methodology and improving the experimental process.
7. **Improvement**: Feedback from validation is incorporated back into the methodology, leading to iterative refinement of models and techniques.

This structured and iterative process ensures the research methodology remains adaptive and effective, ultimately culminating in high-performing models that deliver reliable diagnostic results.

### 3.2 Data Collection and Preprocessing

The foundation of our research was built on the meticulous collection and preprocessing of clinical medical records. Following a formal partnership with a healthcare institution, we acquired a comprehensive dataset comprising 775,000 medical records. These records were anonymized to remove all personally identifiable information (PII) and then processed into a high-dimensional vector space using a Bag of Words (BoW) approach, enhanced by Term Frequency-Inverse Document Frequency (TF-IDF) metrics.

#### 3.2.1 Aggregation of Melanoma Diagnoses

During the preprocessing phase, we implemented a targeted aggregation strategy where all diagnostic categories containing variations of the term ”melanoma” were consolidated into a single ”Melanoma” category. This was achieved by transforming diagnosis labels into lowercase and replacing any occurrence of the term ”melanoma” with ”Melanoma”, thereby facilitating focused analysis on this critical condition.

### 3.3 Neural Network Architecture and Ensemble Method

Our methodology employed an ensemble approach combining two distinct neural network architectures: a Convolutional Neural Network (CNN) and a vanilla Deep Neural Network (DNN). This ensemble method leverages the strengths of both architectures to improve classification accuracy and robustness.

#### 3.3.1 CNN Architecture

The CNN was tailored to capture spatial relationships within the high-dimensional vector representations of the medical records, effectively identifying patterns indicative of various skin diseases.

#### 3.3.2 HSNN Architecture

The core architecture of our model is a hybrid sequential neural network designed for high performance on sequence classification tasks. The model consists of the following layers:

– An **Embedding layer** that transforms each input token into a 100-dimensional vector, capturing semantic similarity among words in a dense representation space.

**Table 2.** Comparison with previous research

| Study | Recall | | Average Recall |
|---|---|---|---|
| | Disease 1 | Disease 2 | |
| Cheng et al. | 82.3% | 85.6% | 83.9% |
| Our Ensemble System | 98.35% (Dermatitis Atopica) | 95.92% (Melasma) | 91.1% |

– A **Spatial Dropout layer** with a dropout rate of 0.2, which helps in reducing overfitting by randomly omitting entire feature maps during training, thus aiding in robust feature learning.
– An **LSTM layer** with 100 units, capable of capturing long-term dependencies in the data, equipped with dropout and recurrent dropout of 0.2 to further aid in preventing overfitting.
– A **Dense output layer** with a softmax activation function, mapping the LSTM outputs to a probability distribution over the target classes, facilitating direct class prediction.

This setup is specifically tailored to process and classify medical text data effectively, where understanding the contextual relationships within clinical notes is crucial.

#### 3.3.3 Integration and Training

The two models were trained on separate partitions of the dataset and their predictions were combined using a weighted voting mechanism. This integration allowed us to harness the predictive power of both models, enhancing the overall accuracy of the system.

### 3.4 Performance Evaluation

We evaluated the model's performance using a variety of metrics including accuracy, precision, recall, and F1-score, specifically focusing on the top three most prevalent diseases identified in the dataset. An extended evaluation was conducted for the newly aggregated "Melanoma" category to assess the effectiveness of our aggregation strategy.

#### 3.4.1 Metrics and Validation

Each model was independently validated using a split-test approach, and the ensemble results were compared against the baseline models to quantify improvements. Additionally, the confusion matrix was employed to visually inspect model performance and misclassifications. The dataset and CNN parameters can be obtained upon request.

## 4 Results

### 4.1 Performance Evaluation

Table 1 summarizes the performance of the ensemble system, which combines the Convolutional Neural Network (CNN) for skin cancer detection and the hybrid sequential neural network (HSNN) for Dermatitis Atopica, Melasma (Cloasma), and Vitiligo classification.

The ensemble system achieves a global F1-score of 95.45%, combining the strengths of both models and ensuring high classification accuracy for all four diseases.

### 4.2 State-of-the-art Comparison

In Table 2, we compare the ensemble system against prior research from Cheng et al., which used a CNN for binary classification of skin cancer. Our system demonstrates superior diagnostic accuracy.

## 5 Conclusions and Future Work

This study confirms the effectiveness of the ensemble system in accurately classifying dermatological diseases from medical records. The combination of CNN and HSNN architectures achieved high precision, recall, and F1 scores across all diseases, offering a robust diagnostic tool. Future Directions: To refine and expand the applicability of this work, we propose:

– Extending the classification framework to include more dermatological conditions, especially rare diseases.
– Integrating text and image data for a comprehensive diagnostic tool, improving diagnostic specificity and sensitivity.

These strategies will enable early detection and personalized treatment of dermatological diseases, reducing misdiagnosis and improving patient outcomes.

## Acknowledgments

The authors wish to express their profound gratitude to the Consejo Nacional de Ciencias, Humanidades y Tecnología (CONAHCYT), acknowledging the institution's significant support, as the main author is favored with a scholarship and two of the authors are affiliated researchers within this esteemed organization. Our deepest gratitude to the Centro de Investigación en Computación of the Instituto Poltécnico Nacional. It is recognized that the synergistic contributions from all involved entities have substantially enhanced the quality of the present study. We also mention that the work was done with partial support from the Mexican Government through the grant A1-S-47854 of CONACYT, Mexico, grants 20241816, 20241819, and 20240951 of the Secretaría de Investigación y Posgrado of the Instituto Politécnico Nacional, Mexico. The authors thank the CONACYT for the computing resources brought to them through the Plataforma de Aprendizaje Profundo para Tecnologías del Lenguaje of the Laboratorio de Supercómputo of the INAOE, Mexico and acknowledge the support of Microsoft through the Microsoft Latin America PhD Award.

## References

1. **Abayomi-Alli, O. O., Damaševičius, R., Misra, S., Maskeliunas, R., Abayomi-Alli, A. (2021).** Malignant skin melanoma detection using image augmentation by oversampling in nonlinear lower-dimensional embedding manifold. Turkish Journal of Electrical Engineering and Computer Sciences, Vol. 29, No. 8, pp. 2600–2614. DOI: 10.3906/elk-2101-133.
2. **Alexiades, M., Zubek, A. (2019).** Cosmetic dermatologic surgery. Google Books.
3. **Ameri, A. (2020).** A deep learning approach to skin cancer detection in dermoscopy images. Journal of biomedical physics & engineering, Vol. 10, No. 6, pp. 801–806. DOI: 10.31661/jbpe.v0i0.2004-1107.
4. **Chambers, D. B., Ghosh, S., Taher, M. S., Salopek, T. G. (2024).** Incidence of nonmelanoma skin cancers in Alberta, Canada, from 2007 to 2018. Journal of Cutaneous Medicine and Surgery, Vol. 28, No. 3, pp. 238–247. DOI: 10.1177/12034754241232677.
5. **Chan, S., Reddy, V., Myers, B., Thibodeaux, Q., Liao, W. (2020).** Machine learning in dermatology: Current applications, opportunities, and limitations. Dermatology and Therapy, Vol. 10, No. 3, pp. 493–511. DOI: 10.1007/s13555-020-00372-0.
6. **Dorj, U. O., Lee, K. K., Choi, J. Y., Lee, M. (2018).** The skin cancer classification using deep convolutional neural network. Multimedia Tools and Applications, Vol. 77, No. 8, pp. 9909–9924. DOI: 10.1007/s11042-018-5714-1.
7. **Fu'adah, Y. N., Pratiwi, N. C., Pramudito, M. A., Ibrahim, N. (2020).** Convolutional neural network (CNN) for automatic skin cancer classification system. IOP conference series: materials science and engineering,

Vol. 982, No. 1, pp. 012005. DOI: 10.1088/1757-899X/982/1/012005.

8. **Gautam, V., Trivedi, N., Anand, A., Tiwari, R., Zaguia, A., Koundal, D., Jain, S. (2023).** Early skin disease identification using deep neural network. Computer Systems Science & Engineering, Vol. 44, No. 3. DOI: 10.32604/csse.2023.026358.
9. **Gouda, W., Sama, N. U., Al-Waakid, G., Humayun, M., Jhanjhi, N. Z. (2022).** Detection of skin cancer based on skin lesion images using deep learning. Healthcare, Vol. 10, No. 7, pp. 1183. DOI: 10.3390/healthcare10071183.
10. **Guimarães, P., Batista, A., Zieger, M., Kaatz, M., Koenig, K. (2020).** Artificial intelligence in multiphoton tomography: Atopic dermatitis diagnosis. Scientific reports, Vol. 10, No. 1, pp. 7968. DOI: 10.1038/s41598-020-64937-x.
11. **Gunwant, H., Joshi, A., Sharma, M., Gupta, D. (2022).** Automated medical diagnosis and classification of skin diseases using EfficientNet-B0 convolutional neural network. New Perspectives on Hybrid Intelligent System Design based on Fuzzy Logic, Neural Networks and Metaheuristics, pp. 3–19. DOI: 10.1007/978-3-031-08266-5_1.
12. **Guo, L., Yang, Y., Ding, H., Zheng, H., Yang, H., Xie, J., Ge, Y. (2022).** A deep learning-based hybrid artificial intelligence model for the detection and severity assessment of vitiligo lesions. Annals of Translational Medicine, Vol. 10, No. 10, pp. 620. DOI: 10.21037/atm-22-1738.
13. **Hammad, M., Pławiak, P., ElAffendi, M., El-Latif, A. (2023).** Enhanced deep learning approach for accurate eczema and psoriasis skin detection. Sensors, Vol. 23, No. 16, pp. 7295. DOI: 10.3390/s23167295.
14. **Kantoria, V., Sharma, S., Bhushan, S., Saini, H., Nijhawan, R. (2020).** Implication of convolutional neural network in the classification of vitiligo. International Research Journal of Engineering and Technology, Vol. 7, No. 3.
15. **Kshirsagar, P., Manoharan, H., Shitharth, S. (2022).** Deep learning approaches for prognosis of automated skin disease. Life, Vol. 12, No. 3, pp. 426.
16. **Kumar, R. S., Singh, A., Srinath, S., Thomas, N. K., Arasu, V. (2022).** Skin cancer detection using deep learning. Proceedings of the International Conference on Electronics and Renewable System, pp. 1724–1730.
17. **Li, Y., Zhao, D., Xu, Z., Heidari, A., Chen, H. (2023).** bSRWPSO-FKNN: A boosted PSO with fuzzy k-nearest neighbor classifier for predicting atopic dermatitis disease. Frontiers in Neuroinformatics, Vol. 16, pp. 1063048.
18. **Li, Z., Koban, K. C., Schenck, T. L., Giunta, R. E., Li, Q., Sun, Y. (2022).** Artificial intelligence in dermatology image analysis: Current developments and future trends. Journal of Clinical Medicine, Vol. 11, No. 22, pp. 6826. DOI: 10.3390/jcm11226826.
19. **Nawaz, M., Nazir, T., Masood, M., Ali, F., Khan, M. A., Tariq, U., Sahar, N., Damaševičius, R. (2022).** Melanoma segmentation: A framework of improved DenseNet77 and UNET convolutional neural network. International Journal of Imaging Systems and Technology, Vol. 32, No. 6, pp. 2137–2153. DOI: 10.1002/ima.22750.
20. **Pangti, R., Mathur, J., Chouhan, V., Kumar, S., Gupta, S. (2021).** A machine learning-based decision support mobile phone application for diagnosis of common dermatological diseases. Journal of the European Academy of Dermatology and Venereology, Vol. 35, No. 2, pp. 536–545. DOI: 10.1111/jdv.16967.
21. **Rasheed, A., Umar, A. I., Shirazi, S. H., Khan, Z., Nawaz, S., Shahzad, M. (2022).** Automatic eczema classification in clinical images based on hybrid deep neural network. Computers in Biology and Medicine, Vol. 147, pp. 105807. DOI: 10.1016/j.compbiomed.2022.105807.
22. **Saifan, R., Jubair, F. (2022).** Six skin diseases classification using deep convolutional neural

network. International Journal of Electrical and Computer Engineering, Vol. 12, No. 3, pp. 3072. DOI: 10.11591/ijece.v12i3.pp3072-3082.

23. **Shivasharan, M. (2020).** Evaluation of accuracy of tibet artificial intelligence application in ten prediction of diagnosis of dermatological conditions. Doctoral dissertation.
24. **Siegel, R. L., Miller, K. D., Wagle, N. S., Jemal, A. (2023).** Cancer statistics, 2023. CA: a cancer journal for clinicians, Vol. 73, No. 3, pp. 233–254. DOI: 10.3322/caac.21601.
25. **Stieler, F., Rabe, F., Bauer, B. (2021).** Towards domain-specific explainable AI: Model interpretation of a skin image classifier using a human approach. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1802–1809.
26. **Sun, N., Chen, B., Zhang, R., Wen, Y. (2022).** Novel neural network model for predicting susceptibility of facial post-inflammatory hyperpigmentation. Medical Engineering & Physics, Vol. 110, pp. 103884. DOI: 10.1016/j.medengphy.2022.103884.
27. **Thomsen, K., Iversen, L., Titlestad, T. L. (2020).** Systematic review of machine learning for diagnosis and prognosis in dermatology. Journal of Dermatological Treatment, Vol. 31, No. 5, pp. 496–510. DOI: 10.1080/09546634.2019.1682500.
28. **Tian, Y., Sun, S., Qi, Z., Liu, Y., Wang, Z. (2022).** Non-tumorous facial pigmentation classification based on multi-view convolutional neural network with attention mechanism. Neurocomputing, Vol. 483, pp. 370–385.
29. **Wei, L., Ding, K., Hu, H. (2020).** Automatic skin cancer detection in dermoscopy images based on ensemble lightweight deep learning network. IEEE Access, Vol. 8, pp. 99633–99647. DOI: 10.1109/ACCESS.2020.2997710.
30. **Zhang, J., Zhong, F., He, K., Ji, M., Li, S., Li, C. (2023).** Recent advancements and perspectives in the diagnosis of skin diseases using machine learning and deep learning: A review. Diagnostics, Vol. 13, No. 23, pp. 3506. DOI: 10.3390/diagnostics13233506.
31. **Zia-Ur-Rehman, M., Ahmed, F., Alsuhibany, S. A., Jamal, S. S., Zulfiqar-Ali, M., Ahmad, J. (2022).** Classification of skin cancer lesions using explainable deep learning. Sensors, Vol. 22, No. 18, pp. 6915. DOI: 10.3390/s22186915.

*Article received on 11/06/2024; accepted on 20/08/2024.*
**Corresponding author is Christian E. Maldonado-Sifuentes.*

ISSN 2007-9737

# Compensatory Fuzzy Logic Genetic Algorithm for Classification Problems: A Case Study

José Fernando Padrón-Tristán[1], Laura Cruz-Reyes[1], Rafael Alejandro Espín-Andrade[2], Carlos Eric Llorente-Peralta[1,*], Fausto Antonio Balderas-Jaramillo[1], Jessica González-San-Martín[1]

[1] Tecnológico Nacional de México,
Instituto Tecnológico de Ciudad Madero,
Division of Graduate Studies and Research,
Ciudad Madero, Tamaulipas,
Mexico

[2] Universidad Autónoma de Coahuila,
Unidad Torreón, Facultad de Contaduría y Administración,
Dirección de Posgrado,
Torreón, Coahuila,
Mexico

jose.pt@cdmadero.tecnm.mx

**Abstract.** This article presents an approach for creating fuzzy predicates using genetic algorithms. The proposed method incorporates an internal genetic algorithm to optimize the membership functions of the linguistic variables involved in the discovered predicates, taking advantage of statistical data for the initialization of the population and taboo and weighted roulettes for the construction of the predicates. The generation of fuzzy predicates is based on the implication and equivalence operators, as well as on deductive structures, such as modus ponens. Furthermore, the evaluation of predicates on data sets is based on For All and Exists quantifier operators, which also guide the search for the best predicates according to their truth values. Furthermore, the popular Iris database is used as a case study to demonstrate the effectiveness and applicability of this approach.

**Keywords.** Compensatory fuzzy logic, genetic algorithms, fuzzy inference, fuzzy interpretability, iris problem.

## 1 Introduction

Fuzzy logic has proven to be a tool for modeling uncertainty and imprecision in complex systems. However, manually creating fuzzy predicates can be laborious and error-prone. In this context, genetic algorithms (GA) have emerged as a technique for optimization and finding solutions to complex problems.

Optimisation is another field where Fuzzy logic stands out because it helps decision-makers to solve optimisation problems considering the uncertainty that commonly occurs in application domains [1].

In this paper, an approach for creating fuzzy predicates using GA is presented, which includes an internal genetic algorithm to optimize the membership functions of linguistic states. Our method is based on the use of statistical data for the initialization of the population and the use of taboo and weighted roulettes for the construction of fuzzy predicates.

Furthermore, the implication and equivalence operators are implemented, as well as the modus ponens deductive structure to compare the accuracy and precision in the classification of the generated fuzzy predicates. Likewise, the parameters of the membership functions that define the optimized linguistic states that are included in the constructed predicates are analyzed. To evaluate and validate this approach, we used the Iris database, which contains trait

measurements of different flower species and which are separated into three different types of this family, as a case study. The experimental results show that the proposed method is capable of generating efficient and accurate fuzzy predicates for flower species classification in the Iris database.

In summary, this article presents a contribution to the field of compensatory fuzzy logic and genetic algorithms, providing techniques for the automated creation of fuzzy predicates with applications in a wide range of real-world problems.

## 2 Fuzzy Logic

Fuzzy logic is a formal system of logic that deals with propositions that can have truth-values, which are intermediate values between false and true, representing degrees of truth. In contrast to classical binary logic, where the result of evaluating a proposition is absolute, whether true or false, fuzzy logic allows for the representation and reasoning about uncertainty and imprecision in data [2].

Fuzzy logic (FL) is based on the idea that the way human thinking is constructed is not through numbers, but rather through linguistic labels. Linguistic terms are inherently less precise than numerical data, but they express knowledge in terms more accessible to human understanding [3, 4].

Vagueness and uncertainty can be considered using the fuzzy set theory proposed by L. Zade [5]. The fundamental concept in fuzzy set theory is the concept of the membership function [6]:

Let $M$ be a set, let $x$ be an element of $M$, then a fuzzy subset A of M is defined as a set of ordered pairs $\{x, \mu A(x)\}, \forall x \in M\ \mu A(x)$ which is a membership function characteristic that takes its values n a well-ordered set $E$, which indicates the degree or level of membership of an element $x$ to a subset $A$.

### 2.1 Compensatory Fuzzy Logic

Compensatory fuzzy logic (CFL) is an approach to multivalent logic different from the axiomatic norm (conjunction) and conorm (disjunction) approaches, which define functions of operations on fuzzy sets. The CFL has characteristics that allow it to be a support for decision-making [7].

The CFL is made up of a quatrain of continuous operators: conjunction (c), disjunction (d), strict fuzzy order (o), and negation (n), where:

$$c[0,1]^n \rightarrow [0,1]$$
$$d[0,1]^n \rightarrow [0,1]$$
$$n[0,1]^n \rightarrow [0,1]$$
$$o[0,1]^n \rightarrow [0,1]$$

These operators satisfy the group of axioms for the FL, to which those of compensation and veto are added [7].

The compensation axiom states that, for the particular case of two components, the fact that the value of the operator is between the minimum and the maximum can be interpreted as the second value compensating the value of the first in the veracity of the conjunction. The idea is generalized to the case of $n$ components [8].

The veto axiom grants any basic predicate of conjunction the ability to veto, that is, the ability to prevent any form of compensation when its value is equal to zero [8].

CFL satisfies the idea that under certain terms compensation is permissible, which is an interesting approach to modeling human decision-making.

### 2.2 Fuzzy Predicates

According to [6], a fuzzy predicate is a function $G(z_1, z_2, \ldots, z_n)$, defined in fuzzy variables $z_1, z_2, \ldots, z_n$ whose range of values is a statement whose truth is estimated by values of the interval (0, 1). This true value is obtained by the composition of fuzzy variables (known as individual fuzzy predicates) and fuzzy logic operators.

### 2.3 Quantifier Operators

The most common way to deal with precision in the inference of fuzzy systems is error-based [9, 10]. To calculate the precision of fuzzy predicates, the universal quantifier ***For All*** is used:

$$\forall_{x,y \in X,Y} = e^{-\sum \ln[P(x,y)]}. \quad (1)$$

Another quantifier used is Exists, which indicates that in a set of data, there are instances for which the evaluated predicate is fulfilled:

$$\exists_{x,y \in X,Y} = e^{-\sum ln[1-P(x,y)]}. \quad (2)$$

In this work, all operator is used to guide the search for the best predicates discovered.

## 3 Fuzzy Inference Systems

Through inference, observations of the world are used to discover unobserved facts or to identify causal effects from the data collected. In a broad sense, inference ranges from implication to the operational mental process that allows reaching a conclusion based on certain information [11].

From formal logic, inference has been understood as "the passage from one set of propositions to another; the first set can be called the class of premises and the second the class of conclusions." This definition presents inference as a syllogistic structure, that is, a three-level logical entity: two premises and a conclusion. Thus, inference not only reveals unobserved facts, but also follows a logical structure that facilitates the derivation of conclusions from given premises [12].

### 3.1 Mamdani Fuzzy Inference System

The Mamdani fuzzy inference system (FIS), introduced by Ebrahim Mamdani in 1975, stands as one of the earliest and most prevalent systems in fuzzy logic. It serves as a technique for framing control issues using fuzzy logic principles, emulating human decision-making processes. This model is renowned for its simplicity and its ability to intuitively encapsulate expert Knowledge [13].

This model is characterized by its simplicity and the intuitive way it represents expert knowledge. Stages of the fuzzy inference system Mamdami based are (figure 1):

1 Fuzzification: crisp input variables are converted into fuzzy sets through a MF. For each input is obtained how much belongs to a fuzzy set.

2 Rule Evaluation: The system uses a set of if-then rules that describe how input values translate into outputs. These rules are expressed in terms of fuzzy logic.

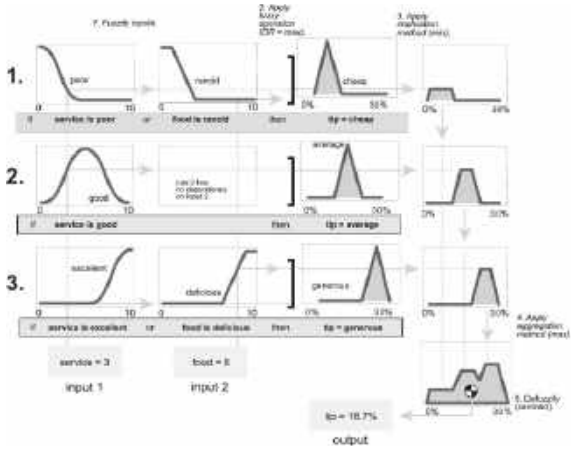



**Fig.1.** Stages of the Mamdani's FIS [14]

3 Aggregation of Rule Outputs: Rules are evaluated and their fuzzy outputs are combined into a single fuzzy output set.

4 Defuzzification: The fuzzy output is converted back into a crisp value. Common defuzzification methods include the centroid, weighted average, and maximum height methods.

### 3.2 Takagi-Sugeno FIS

The Sugeno fuzzy inference model, also referred to as the Takagi-Sugeno-Kang (TSK) model, was developed by Takagi and Sugeno in 1985. In contrast to the Mamdani model, which produces fuzzy set outputs for its rules, the Sugeno model generates outputs that are linear functions of the inputs [15].

Stages of the TSK model are listed next (figure 2):

1 **Fuzzification**: Similar to the Mamdani model, crisp inputs are converted into fuzzy sets using membership functions.

2 **Rule Evaluation**: The if-then rules in the Sugeno model are of the form: "If x is A and y is B, then z = f(x, y)," where f(x,y) is a polynomial function.

3 **Aggregation of Rule Outputs**: The rule outputs are crisp functions (linear or constant) and are aggregated through a weighting process.

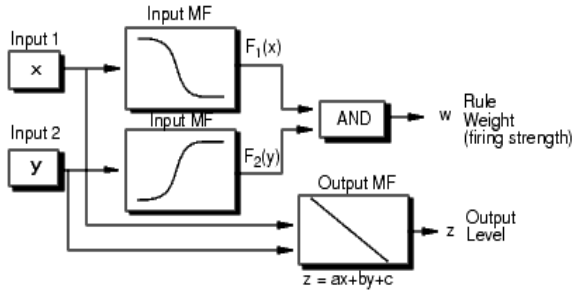



**Fig. 2.** TSK model [14]

4 **Defuzzification**: Instead of defuzzifying a fuzzy set, the Sugeno model produces a crisp output directly by the weighted combination of the rule outputs.

### 3.3 Comparison of Models Mamdani with TSK

Table 1 shows a comparison of the Mamdani model with the Sugeno model:

Analyzing the advantages and disadvantages that appear in both models, it was decided to work with the Mamdani model, since it presents the following characteristics [13, 5]:

1 **Intuitive rule representation:** The rules in the Mamdani model are formulated in natural language, such as "if x is A and y is B then z is C". This structure closely mirrors human reasoning, making the system more understandable for domain experts who may not have a background in mathematics.

2 **Richness of output expression:** The Mamdani model produces outputs as fuzzy sets, enabling a more nuanced and expressive depiction of uncertainty and partial truths.

3 **Ease of rules definition and maintenance:** Since the rules are based on linguistic terms that directly correspond to expert knowledge and real-world observations, the system is easier to update and expand.

4 **Suitable for complex systems with multiple inputs and outputs:** This feature makes the model ideal for applications where the relationships between variables are difficult to quantify.

## 4 Classification Task

In this work binary classification is performed, which means that the GA generates a predicate that determines if an instance in the dataset belongs to a category or not.

If the instance belongs to the searched category, then is called a positive instance, otherwise it is considered a negative case.

GA discovers the best predicates according to their truth value and is executed once per class in the dataset. If the dataset contains $m$ classes, then the GA is performs $m$ times the discovery of predicates to find the best ones corresponding to each class. To do this, the corresponding $m$ symbolic predicates are created as part of the GA configuration, which are listed as follows:

$$(IMP\ p_1(x)\ class_1)$$
$$(IMP\ p_2(x)\ class_2)$$
$$\vdots$$
$$(IMP\ p_m(x) m\ class_m).$$

For each instance, the classification task takes the best predicate according to its truth value to extract the premise or the generated sub-predicates $p_j(x)$ to be evaluated and compared with the others.

The $p_j(x)$ with the maximum truth value, is selected to get the class obtained for the instance.

The accuracy for the classification task is given for [16]:

$$Acc = \frac{tp+tn}{tp+fp+tn+fn}, \quad (3)$$

where:

$tp$: (true positive) positive instances correctly classified.

$tn$: (true negative) negative instances correctly classified.

$fp$: (false positive) instances incorrectly classified as positives.

$fn$: (false negatives) instances incorrectly classified as negatives.

## 5 State of the Art

Data classification is a widely researched area in pattern recognition. It involves dividing the data space, known as pattern space, into different classes. Among various applications, there are: image processing, classification and segmentation; voice recognition and restoration; signal processing and, among others, financial data analysis; human action recognized in video [17]; facial geometry identification [18].

The design of a classifier is based on a set of data divided into categories, which comprises data vectors along with their expected labels, known as labeled data [19].

A well-designed classifier must have a property known as generalization, which is to accurately classify new and unseen data with a low error rate [19]. To this end, several methods have been used for several years, including those based on distance, statistical methods, neural networks, and fuzzy logic (FL) [20].

FL was introduced by Lofti A. Zadeh [5, 21], which uses membership functions to model knowledge. Membership functions quantify the degree to which a feature of data satisfies an attribute, with truth values ranging between 0 and 1 in FL.

Data acquisition has improved and accelerated through technological advances in recent decades. Likewise, improvements in hardware have been used to propose more computationally intensive algorithms for data processing. But despite the successes obtained in pattern recognition, approaches are required that facilitate the discovery of knowledge and allow the analysis of the extensive information contained in the data [22].

An approach that has gained strength in recent years is the automation of the definition of membership functions (MF) and the generation of predicates from a data set [23, 24] This approach allows the analysis of MFs and predicates eliminating the need for expert knowledge in the use of FL.

Unlike the traditional way, in which an FL model was defined based on expert knowledge, it is now derived automatically from the data, this being a crucial aspect. According to Zadeh [25], MFs and predicates encapsulate the meanings of expressions in natural language, making their interpretation closely linked to knowledge. Preserving the semantic meaning of natural language expressions to achieve interpretable descriptions is essential; Otherwise, pattern recognition-based methods could supplant FL-based models.

**Table 1.** Comparison of models Mamdani and TSK

| Attribute | Mamdani | Sugeno |
|---|---|---|
| Efficient | Less efficient | More efficient |
| Computational Efficiency | Intense | Reduced computational cost |
| Precision | Limited precision | Higher precision |
| Intuitiveness | Easy to understand; accessible to domain experts | More difficult to understand; for experts with mathematical training |
| Flexibility | Can handle systems with multiple inputs and outputs | Less flexible; to define the output functions can be difficult |

On the other hand, the use of the CFL presents a series of advantages such as the following [7]:

Compensation in fuzzy predicate evaluations: If a value does not completely meet one criterion, but largely meets another, these values can compensate for each other, providing a more balanced and realistic approach in modeling complex systems.

Adaptability: can better adjust to the nuances of the relationships between variables.

More informed decisions: Provides a stronger basis for decision making, especially in situations where decision criteria are conflicting or are not equally important.

Flexibility: reduce the inflexibility of decisions based solely on strict individual criteria.

In addition to these advantages, in this work the generalized membership function [10] (GMF, section 6.2) was used, which is a flexible function, with which any linguistic state can be represented, since it can take a sigmoidal, sigmoidal form.

negative, Gaussian or concave, based on its parameters.

In this work, techniques were also explored to control the length and consistency of fuzzy predicates and types of predicates for the classification of instances in a data set were also explored, such as those that use the implication operator, the equivalence and also, the deductive structure Modus Ponens, complementing it with the existential quantifier.

## 6 Proposal of Solution

The objective of this work is to prove that a genetic algorithm that uses CFL provides a solid methodology that allows for solving classification problems through fuzzy predicates.

To achieve this objective there is a dataset that contains a register of 150 instances of iris flower and measurements of sepal and petal of the flower. This dataset and a set of parameters of configuration are the input of a genetic algorithm, which implements a series of techniques to generate predicates that classify what type of iris is a flower according to its attributes.

1 The dataset and a configuration set are the input of the GA, including a symbolic predicate that establishes what logical operator of equivalence or implication or a deductive structure is going to be used.
2 The GA creates fuzzy predicates according to the input optimizing the parameters of the membership functions.
3 The GA outputs a list of the best predicates according to their truth value. GA is executed for each class in the dataset (setosa, virginica, and versicolor) and, for each symbolic predicate (implication and equivalence operators and deductive structure modus ponens).
4 Sub-predicates are extracted from generated fuzzy predicates to evaluate each instance in the dataset.
5 For each instance, the truth value of the sub-predicates corresponds to the implication, equivalence, and deductive structure modus ponens. Through the highest, the class for the instance is selected.
6 The accuracy is calculated according to equation (3) and compared with the other proposals to get the highest.

## 7 Genetic Algorithm

The proposed genetic algorithm (GA) includes techniques to maximize the truth-value of results, such as:

1 Multiple generators of sub-predicates with independent search configuration in the creation of chromosomes.
2 Linguistic variables are defined by the generalized membership function.
3 Initial population based on statistical data.
4 Parameters optimization through an intern genetic algorithm.
5 Taboo roulette to avoid inconsistent predicates.
6 Weighted roulette to control predicate length.
7 Calculation of universal and existential quantifiers in fuzzy predicates.
8 Use of correct deductive structures as an objective function.

The genetic algorithm is given for the next steps (Algorithm 1):

The initial population is generated based on statistical data in the dataset, and then according to the total generations set in the configuration, the genetic operators: selection, crossover, mutation, optimization, and reduction of the population of chromosomes are performed.

In the generation of initial population and selection, the taboo and probabilistic roulettes are used.

The optimization of chromosomes is the generation of the best parameters of GMF of linguistic variables and it is performed through an internal GA.

**Algorithm 1. Genetic Algorithm**

Input:
Search Algorithm configurations
D: Dataset
Output:
R: Set of best fuzzy predicates based on truth-value

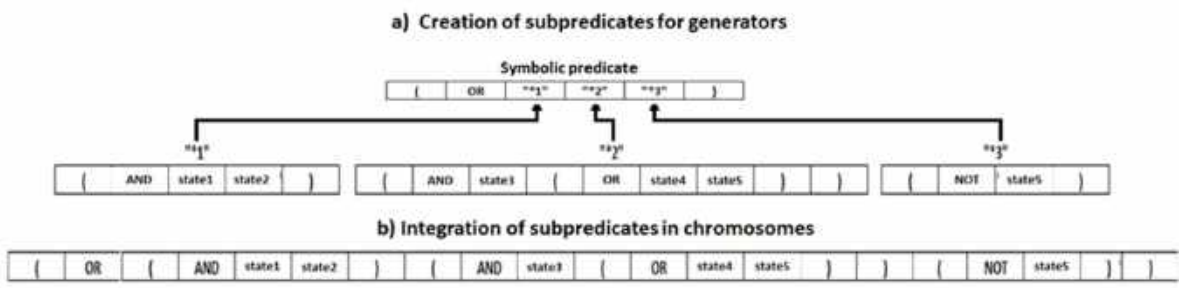



**Fig. 3.** Integration of sub-predicates in chromosome

```
R=Generate Initial Population
For i=1 to total Generations
        parents=selection(R)
        R=crossover(R,parents)
        R=mutation(R)
        R=optimize(R)
        R=reduce(R)
R=sort(R)
return R
```

The generation of initial population (step 1) is done through the creation of chromosomes and the initialization of the parameters of the MF defining the linguistic variables that are included (sections 6.1 - 6.3). This operation also uses the taboo roulette to avoid the generation of inconsistent sub-predicates (Section 6.8).

The selection operator (step 3) generates the parents matrix, which contains $n$ rows and 2 columns to form pairs of chromosomes that are chosen to interchange genes. Selection operator works as follows:

1 The first parent is the best chromosome available according to its truth value.
2 The second parent is chosen through the weighted roulette, which provides a greater probability of being selected to the chromosomes with the best truth value.
3 This operations are repeated to complete the required $n$ pairs of chromosomes.
4 The crossover, mutation and optimization operators are explained in sections 6.4 - 6.6 respectively.
5 The population of chromosomes is reduced by a percentage according to the GA configuration.
6 After calculating the total number of generations, the population is ordered according to the truth value of the chromosomes.

### 7.1 Chromosomes

In the GA, which is a search algorithm, chromosomes are a representation of fuzzy predicates, which are evaluated over the dataset to calculate the truth value.

The GA is configured through a list of parameters to generate fuzzy predicates that include (figure 3):

1 A symbolic predicate that defines the structure of the searched predicates, i. e.:

$$(OR\text{“}*1\text{”}\text{“}*2\text{”}\text{“}*3\text{”}).$$

2 Depth defines how many levels maximum is in the predicate.
3 Logical operators included.
4 Linguistic variables, among others.

Symbolic predicates may include generators of sub-predicates, which are represented by stars and a consecutive number. These generators have an independent configuration with their parameters for the search algorithm (table 2).

According to their configuration, generators create sub-predicates which include a set of parameters (Figure 3). 1=Generator, 2=Linguistic variables, 3=depth, 4=Logical operators.

Algorithm 2 shows how are created the subchromosomes, a fragment of a chromosome corresponding to a generator:

**Algorithm 2. Create subChromosome**

Input:

$v$: List of linguistic variables
$o$: List of logical operators
$c$: Search configuration

Output:

$s = \{\emptyset\}$: subchromosome

1. $n = roulette.getValue$(depth) // return the amount of levels for the subredicate in [0, depth]
2. $if\ n = 0$
3. $index = tabooRoulette(v)$
4. $s = s \cup v_{index}$
5. $return\ s$
6. $index = roulette.getValue$(o)
7. $operator = o_{index}$
8. $s = s \cup \{(", \text{operator}. ")\}$
9. $for\ level = 1\ to\ total\ n$
10. $for(index = 1\ to\ size\ of\ s$
11. currentLevel = $searchInCurrentLevel(index, level, s)$
12. $If\ currentLevel = true$
13. $s_{index} = insertOperand(s, index, v, tabooRoulette)$
14. $return\ s$

The method searchInCurrentLevel verifies that element i corresponds to the current depth level within the subpredicate and returns true if so and false otherwise.

The insertOperando method looks for the index element in s and inserts an operator or a linguistic variable, which are selected randomly.

If an operator is selected, then:

$s = \{s_1, s_{index}\} \cup \{(",\text{operator}, ")"\} \cup \{s_{index+1}, s_n\},$

where operator is selected randomly.

If a linguistic variable is selected, then:

$s = \{s\_1, s\_index\} \cup v\_i \cup \{s\_(index + 1), s\_n\},$

where

$$i = tabooRoulette(v).$$

### 7.2 Linguistic Variables

The linguistic states are presented as linguistic variables initially, in which the parameters of the membership functions (MF) are going to be optimized to maximize the truth value of the predicate. The MF used to define the linguistic variables is the generalized membership function GMF [10]:

$$GMF = \frac{sigm(x,\gamma,\alpha)^m[1-sigm(x,\gamma,\alpha)]^{1-m}}{M}. \quad (4)$$

With $sigm(x, \gamma, \alpha) = \frac{1}{1+e^{-\alpha(x-\gamma)}}$.

Here $M = m^m(1-m)^{1-m}$ and

1 $\gamma$ defines the center of the graphic of the function,

2 $\alpha$ defines the breadth of the function,

3 $m$ modifies the form of the function.

The advantage of the GMF over other membership functions is that it has the flexibility to change the shape of the function and can be used to model different linguistic variables, unlike other functions such as triangular or trapezoidal that require other types of functions to model different linguistic states (figure 4).

**Table 2.** Configuration of the search algorithm

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| *1 | state1, state2, state3, state 5 | 1 | AND |
| *2 | state3, state4, state5 | 2 | AND, OR |
| *3 | state1, state3, state5 | 1 | NOT |

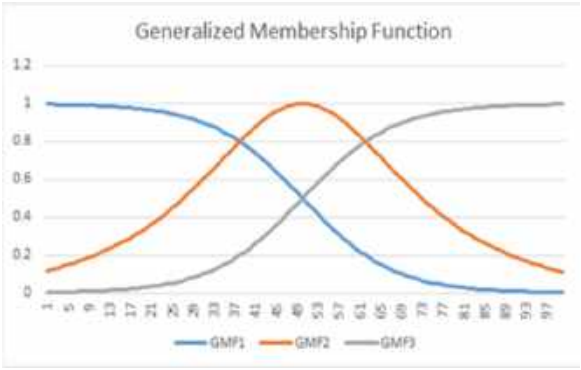



**Fig. 4**. GMF with parameters m=0 (GMF1), m=0.5 (GMF2) and, m=1 (GMF3)

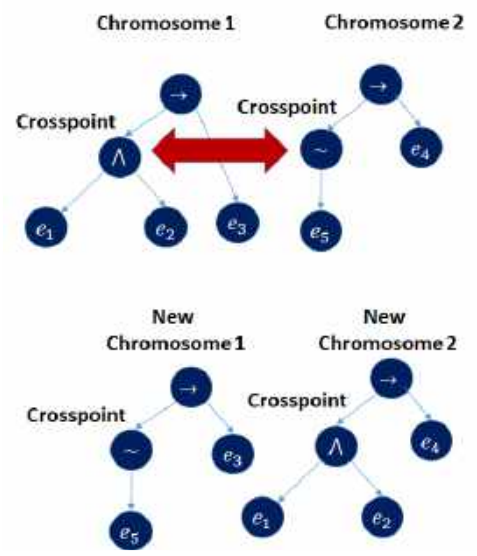



**Fig. 5.** Crossover operation

### 7.3 Membership Function's Parameters Initialization

In the discovery of fuzzy predicates, the first data is separated into two parts by attributes. The first part includes the attributes of the condition, meaning all of them will be used to create sub-predicates that correspond to the attribute of decision, which is the second part of attributes in the dataset.

For each attribute in the dataset, it is calculated the values of the parameters through percentiles and according to the number of linguistic variables associated with each attribute.

For each variable, the gamma parameter is calculated according to the percentile that corresponds to it, for example, if there are three linguistic variables for an attribute, the corresponding percentiles are 25, 50 and 75 and the gamma values are those that correspond to these percentiles.

The dataset is normalized, and all attributes of the condition and all instances take values in the range [0, 1]. This normalization allows the parameters of the GMF $\gamma$ and $m$ to take values also in [0, 1] and $\alpha$ is standardized in [10, 50].

According to the linguistic variable $i$ defined for the attribute of condition $k$, algorithm 3 shows the definition of each linguistic variable *s*:

**Algorithm 3. Definition of GMF**

Input
$D$: Dataset normalized
Output
$E$: Set of linguistic variables
1. Let $A$ be the set of attributes in $D$
2. For each $a \in A$
3. Let $n_k$ the number of linguistic variables for $a$
4. For $i = 1\ to\ n_k$
5. $e_{label} = a_{name} + i$
6. $e_{attribute} = a_{name}$
7. $e_{MF} = "GMF"$
8. $e_\gamma = noise\left(calcPercentile\left(a, \frac{100i}{n_k+1}\right)\right)$
9. $e_\alpha = noise(35\ rand(0,1) + 15)$
10. $e_m = noise\left(\frac{m_i - m_1}{m_{n_k} - m_1}\right)$
11. $E = E \cup e$
12. return $E$

The method noise injects random variations in range [-0.05,0.05] to the computed parameters.

### 7.4 Crossover Operator

In the crossover operator, the chromosomes selected as parents generate copies of themselves and then the created chromosomes exchange information. On each chromosome created, a crossover point is selected, which can point to a linguistic variable or a complete sub-predicate (Figure 5).

The linguistic variable or the corresponding sub-predicate is taken from each chromosome and is integrated into the other at the crossing point that corresponds to each one.

It is validated that the new chromosomes comply with the restrictions of the GA search algorithm configuration; if not, the operation is repeated.

If they comply with the configuration, the truth value of the altered chromosomes is calculated and they are integrated into the population.

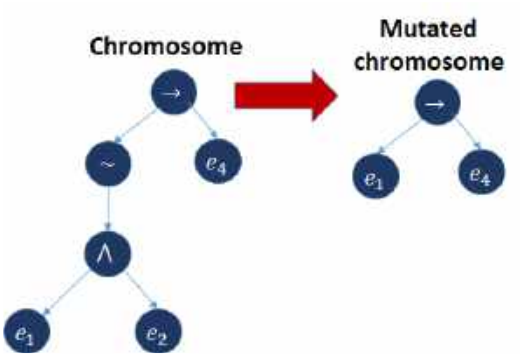



**Fig. 6.** Mutation operation

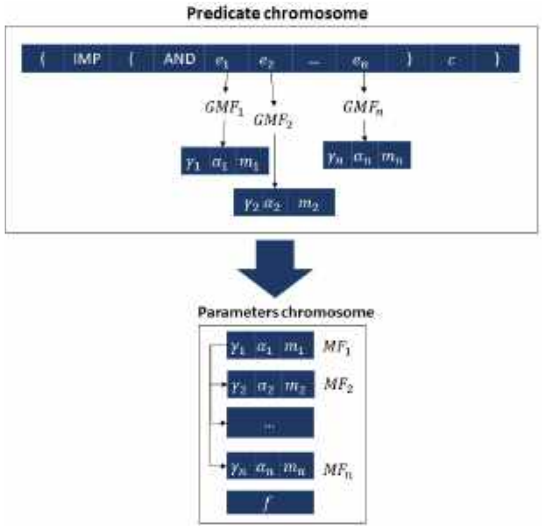



**Fig. 7.** Extraction of the parameters of the predicate to create the parameter's chromosome

### 7.5 Mutation Operator

In the mutation operator, a chromosome is selected to be altered. On that chromosome, a point is randomly selected at which a linguistic variable or a sub-predicate can be found (figure 6).

This variable or sub-predicate is extracted from the chromosome and is replaced by an optimized sub-predicate or variable.

It is validated that the new chromosome meets the restrictions of the search algorithm configuration and if it does not, this operation is repeated.

If the new chromosome meets these constraints, then its truth value is calculated and it is returned to the population.

### 7.6 Optimization of Parameters in Membership Functions

The optimization of parameters for GMF is done through an internal GA that first extracts the parameters to build a new population of chromosomes that work with $\alpha, \gamma,$ and $m$ related to the linguistic variables in the fuzzy predicates (Figure 7, steps 1 and 2 of algorithm 4). In the generation of the initial population, the parameters of the MF are injected with noise.

The algorithm 4 shows how genetic operators such as selection, crossover, and mutation are applied to the population of parameters' chromosomes for the number of generations set in the GA configuration.

In this algorithm, whenever a chromosome is altered, the parameters in it are substituted into the predicate to obtain the truth value of the chromosome.

The selection operator works in the same way as in algorithm 3.

In the crossover operator, the parent chromosomes create copies of themselves and in them, they go through parameter by parameter and exchange the values randomly. If exchanges could not be carried out, the operation is repeated.

In the mutation operator, the selected chromosome is traversed parameter by parameter to alter some of them by selecting them randomly.

Finally, the fuzzy predicate is returned with the set of parameters that maximize the truth value in it.

**Algorithm 4. Parameter Optimizer**

Input

$D$: Dataset normalized

$p$: Fuzzy predicate

Output

$q$: Predicate with best set of parameters of MF according to their truth value

1. $r = extractParameters(p)$
2. $S = generateInitialPopulation(r)$
3. For $i = 1$ $to$ $totalGenerations$
4. $\quad parents = selection()$
5. $\quad S = crossover(S, parents)$
6. $\quad S = mutation(S)$
7. $return\ q = \max_{truthvalue} S$

### 7.7 Weighted Roulette

Logical operators have different arity (Table 3). In the case of conjunctions and disjunctions, to avoid predicates with a high number of linguistic states that increase the complexity of their reading, a weighted roulette is proposed that assigns a greater probability of choosing a smaller number of linguistic states in the operation.

Given the following: If $r_i = q^{i-2}$ where $q = 0.5 < i < 1$ and $i = 2\ to\ n$ is the number of operands, then the cumulative probability that the selected operator has $i$ linguistic states as operands is given by:

$$p_i = \frac{r_i}{\sum_{i=2}^{n} r_i} + \begin{cases} 0 & if\ i = 2, \\ p_{i-1} & if\ i > 2, \end{cases} \quad (5)$$

with i = 2 to n.

### 7.8 Taboo Roulette

Taboo roulette is implemented to avoid inconsistent predicates, which are those that in the logical operations contain duplicate linguistic variables in the case of implication, equivalence, or disjunction operators, for example:

$$(IMP \text{ high chlorideshigh chlorides})$$

$$(OR\ "\boldsymbol{high\ alcohol}"\ "\boldsymbol{high\ alcohol}"$$

$$"low\ pH")$$

In the case of conjunctions, inconsistent predicates, besides incorporating duplicate linguistic states, also can include contradictory linguistic states, that is, they correspond to a fuzzy partition and generate contrary meanings that result in low truth values. An example is the following:

$$("AND"\ "low\ chlorides"\ "high\ chlorides").$$

In disjunctions, to avoid the generation of contradictory predicates in the initial population, a taboo roulette is used, in which in the list of participating slots, once a linguistic state is selected, it is eliminated from the roulette and added to a taboo list. This will be done for each logical operation in the predicate.

**Table 3.** Operators' arity

| Operator | Arity |
|---|---|
| Negation (NOT) | 1 |
| Implication (IMP) | 2 |
| Equivalence (EQV) | 2 |
| Conjunction (AND) | 2 to n |
| Disjunction (OR) | 2 to n |

In conjunctions, when a linguistic state is selected, all states with the corresponding attribute are taken to the taboo list.

Once the selection of linguistic states in a sub-predicate is completed, the roulette wheel restarts, returning all states in the taboo list to the list of participating slots.

Each slot includes:

1. Name of the linguistic state
2. Attribute to which the linguistic state belongs in the data set.

### 7.9 Calculation of Universal and Existential Quantifiers in Fuzzy Predicates

In this genetic algorithm it was implemented the evaluation of quantifiers included in the fuzzy predicate, to calculate the truth value of the quantifiers, first, they are evaluated and the result obtained replaces the partial operation in the predicate, for example:

$(AND\ \ (EXIST\ "sepal\text{-}length")$
$(FORALL\ \ "sepal\text{-}width")\ petal\text{-}length).$

First, evaluate $(EXIST"sepal\text{-}length")$, then evaluates $(FORALL"sepal\text{-}width")$ and the resulting predicate remains as follows:

$$(AND\ 0.5789\ 0.3724\ "petal\text{-}length").$$

The resulting predicate can now be evaluated for the instances in the dataset.

### 7.10 Use of Correct Deductive Structures as the Objective Function

The implication operator has a combination that results in ineffective like the following:

$$False \rightarrow True \Longrightarrow True.$$

In CFL, the combination of:

$$low\ truth\ value \longrightarrow high\ truth\ value \Longrightarrow high\ truth\ value.$$

Which means that a premise with a low truth value implies a conclusion with a high truth value, which is not correct, that is, you can have a predicate that evaluated can indicate that an instance corresponds to a result or a class to which is not related.

For this reason, it was implemented a correct deductive structure as an objective function such as modus ponens, which is combined with the existential quantifier. In this way, the objective function is defined like this:

$$\exists P \bigwedge (P \rightarrow Q).$$

The predicate notation in the algorithm is as follows:

$$\big(AND\,(EXIST\ *1")\ (IMP\ *1"\ "class")\big).$$

## 8 Experimentation

As a case study, the Iris set was used for experimentation. The data set contains 50 samples of each of the three Iris species (setosa, virginica, and versicolor). Four traits were measured for each sample: the length and width of the sepal and petal, in centimeters.

The data set with 150 records was divided into two parts, one for discovery and the other for inference, each part with 120 and 30 respectively.

The data set was used for the classification task, where a class was assigned for each type of Iris species and, this attribute is defined as the decision variable within the classification.

On the other hand, the length and width attributes of the sepal and petal are defined as the condition variables, that is, those by which it is possible to find which type of Iris each instance in the data set corresponds to.

The following configuration of the search algorithm was used for the discovery of fuzzy predicates (Table 4):

The following are symbolic predicates:

1 The implication

$$(IMP\ "*1"\ \ \text{setosa}"),$$

2 The equivalence

$$(EQV\ "*1"\ \ \text{setosa}"),$$

3 The modus ponens deductive structure

$$(AND\ (EXIST\ "*1")(IMP\ "*1"\ \ \text{setosa})).$$

**Table 4.** Configuration of GA

| Parameter | Value |
|---|---|
| Máximum Results | 10 |
| Total generations | 10 |
| Minimum truth-value | 0.8 |
| Mutation percentage | 0.05 |
| Initial Population | 50 |
| Depth | 1 |
| Máximum Results | 10 |

In the same way, the symbolic predicates for the classes *virginica* and *versicolor*.

To calculate the accuracy of this work it was used k-cross-validation and to get the number of folds it was calculated by:

$$K = \ln(total\ instances) = \ln(150) = 5.01 = 5.$$

## 9 Results

Table 5 shows the best predicates obtained for each class using the implication operator, in the same way, tables 6 and 7 show the best predicates using equivalence and correct deductive structure, respectively.

Sub-predicates that act as a premise are extracted from each predicate and evaluated to get its truth value for each instance.

For equivalence, the sub-predicates extracted are:

1 $(AND\ \text{petal\_width}\text{petal\_length}),$
2 $(AND\ \text{sepal\_length}\text{petal\_width}\ "petal\_length"),$
3 $(AND\ \text{sepal\_length}\text{petal\_length}).$

**Table 5.** Best fuzzy predicates obtained for each class for implication operator

| Class | Predicate |
|---|---|
| Setosa | *(EQV (AND "petal_width" "petal_length") "setosa")* |
| Virginica | *(EQV (AND "sepal_length" "petal_width" "petal_length") "virginica")* |
| Versicolor | *(EQV (AND "sepal_length" "petal_length") "versicolor")* |

**Table 6.** Best fuzzy predicates obtained for each class for implication operator

| Class | Predicate |
|---|---|
| Setosa | *(IMP (AND "petal_width" "petal_length") "setosa")* |
| Virginica | *(IMP (AND "sepal_length" "petal_width" "petal_length") "virginica")* |
| Versicolor | *(IMP (AND "sepal_length" "petal_length") "versicolor")* |

**Table 7.** Best fuzzy predicates obtained for each class for correct deductive structure

| Class | Predicate |
|---|---|
| Setosa | *(AND (EXIST (AND "petal_length" "sepal_width") (IMP (AND "petal_length" "sepal_width") "setosa"))* |
| Virginica | *(AND (EXIST (AND "sepal_length" "sepal_width" "petal_length" "petal_length") (IMP (AND "sepal_length" "sepal_width" "petal_length" "petal_length") "virginica"))* |
| Versicolor | *(AND (EXIST (AND "sepal_length" "petal_width") (IMP (AND (EXIST (AND "sepal_length" "petal_width") "versicolor"))* |

**Table 8.** Fragment of results of classification task for equivalence operator

| RC | Se | Vi | Ve | OC | Cl |
|---|---|---|---|---|---|
| 1 | **0.998** | 0.086 | 0.000 | 1 | Hit |
| 3 | 0.000 | 0.287 | **0.622** | 3 | Hit |
| 3 | 0.000 | 0.142 | **0.912** | 3 | Hit |
| 2 | 0.000 | **0.635** | 0.079 | 2 | Hit |
| 2 | 0.000 | **0.603** | 0.202 | 2 | Hit |
| 2 | 0.000 | **0.635** | 0.125 | 2 | Hit |
| 1 | **0.991** | 0.050 | 0.000 | 1 | Hit |
| 1 | **0.994** | 0.091 | 0.000 | 1 | Hit |
| 1 | **0.987** | 0.070 | 0.000 | 1 | Hit |
| 3 | 0.000 | 0.342 | **0.601** | 3 | Hit |

For implication:

1. ($AND$ "$petal\_width$" "$petal\_length$"),
2. ($AND$ $sepal\_length$ "$petal\_width$" "$petal\_length$"),
3. ($AND$ "$sepal\_length$" "$petal\_length$").

For the correct deductive structure:

1. ($AND$ "$petal\_length$" "$sepal\_width$"),
2. ($AND$ sepal_lengthsepal_width "$petal\_length$" "$petal\_length$")

3 ($AND$ "$sepal\_length$" "$petal\_width$")

**Table 9.** Classification results for an equivalence operator

| Fold | Setosa | | | | | | Virginica | | | | | | Versicolor | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN |
| 1 | 11 | 19 | 11 | 19 | 0 | 0 | 11 | 19 | 11 | 18 | 1 | 0 | 8 | 22 | 7 | 22 | 0 | 1 |
| 2 | 8 | 22 | 8 | 22 | 0 | 0 | 12 | 18 | 12 | 16 | 2 | 0 | 10 | 20 | 8 | 20 | 0 | 2 |
| 3 | 10 | 20 | 10 | 20 | 0 | 0 | 7 | 23 | 7 | 18 | 5 | 0 | 13 | 17 | 8 | 17 | 0 | 5 |
| 4 | 9 | 21 | 9 | 21 | 0 | 0 | 11 | 19 | 11 | 15 | 4 | 0 | 10 | 20 | 6 | 20 | 0 | 4 |
| 5 | 12 | 18 | 12 | 18 | 0 | 0 | 9 | 21 | 8 | 20 | 1 | 1 | 9 | 21 | 8 | 20 | 1 | 1 |

**Table 10.** Classification results for implication operator

| Fold | Setosa | | | | | | Virginica | | | | | | Versicolor | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN |
| 1 | 11 | 19 | 11 | 19 | 0 | 0 | 11 | 19 | 11 | 18 | 1 | 0 | 8 | 22 | 7 | 22 | 0 | 1 |
| 2 | 8 | 22 | 8 | 22 | 0 | 0 | 12 | 18 | 12 | 16 | 2 | 0 | 10 | 20 | 8 | 20 | 0 | 2 |
| 3 | 10 | 20 | 10 | 20 | 0 | 0 | 7 | 23 | 7 | 18 | 5 | 0 | 13 | 17 | 8 | 17 | 0 | 5 |
| 4 | 9 | 21 | 9 | 21 | 0 | 0 | 11 | 19 | 11 | 15 | 4 | 0 | 10 | 20 | 6 | 20 | 0 | 4 |
| 5 | 12 | 18 | 12 | 18 | 0 | 0 | 9 | 21 | 8 | 20 | 1 | 1 | 9 | 21 | 8 | 20 | 1 | 1 |

**Table 11.** Classification results for the correct deductive structure

| Fold | Setosa | | | | | | Virginica | | | | | | Versicolor | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN | P | N | TP | TN | FP | FN |
| 1 | 12 | 18 | 12 | 18 | 0 | 0 | 7 | 23 | 6 | 18 | 5 | 1 | 11 | 19 | 6 | 18 | 1 | 5 |
| 2 | 10 | 20 | 10 | 20 | 0 | 0 | 9 | 21 | 9 | 16 | 5 | 0 | 11 | 19 | 6 | 19 | 0 | 5 |
| 3 | 7 | 23 | 7 | 23 | 0 | 0 | 12 | 18 | 11 | 13 | 5 | 1 | 11 | 19 | 6 | 18 | 1 | 5 |
| 4 | 7 | 23 | 7 | 23 | 0 | 0 | 16 | 14 | 15 | 12 | 2 | 1 | 7 | 23 | 5 | 22 | 1 | 2 |
| 5 | 14 | 16 | 14 | 16 | 0 | 0 | 6 | 24 | 5 | 17 | 7 | 1 | 10 | 20 | 3 | 19 | 1 | 7 |

**Table 12.** Accuracy for equivalence operator

| F | Accuracy | | | |
|---|---|---|---|---|
| | Setosa | Virginica | Versicolor | Av |
| **1** | 100% | 97% | 97% | 98% |
| **2** | 100% | 93% | 93% | 96% |
| **3** | 100% | 83% | 83% | 89% |
| **4** | 100% | 87% | 87% | 91% |
| **5** | 100% | 93% | 93% | 96% |
| | | | **Total** | **94%** |

**Table 13.** Accuracy of classification for implication operator

| F | Accuracy | | | |
|---|---|---|---|---|
| | **Setosa** | **Virginica** | **Versicolor** | **Av** |
| **1** | 97% | 50% | 53% | 67% |
| **2** | 100% | 80% | 80% | 87% |
| **3** | 100% | 67% | 67% | 78% |
| **4** | 100% | 63% | 63% | 76% |
| **5** | 100% | 63% | 63% | 76% |
| | | | **overall accuracy** | **76%** |

**Table 14.** Accuracy of classification for deductive structure modus ponens

| F | Accuracy | | | |
|---|---|---|---|---|
| | **Setosa** | **Virginica** | **Versicolor** | **Av** |
| **1** | 100% | 80% | 80% | 87% |
| **2** | 100% | 83% | 83% | 89% |
| **3** | 100% | 80% | 80% | 87% |
| **4** | 100% | 90% | 90% | 93% |
| **5** | 100% | 63% | 63% | 76% |
| | | | **overall accuracy** | **76%** |

**Table 15.** Comparison of accuracy of operators and CDS Modus Ponens

| **Proposal** | **Accuracy** |
|---|---|
| Equivalence | **94 %** |
| Implication | 76 % |
| Modus Ponens | 88 % |

Respective truth values are compared to get the highest, which indicates the selected class for each instance (table 8).

RC=Real class, $Se, Vi, Ve$ = truth-value for setosa, virginica, and versicolor, in bold the highest truth-value, OC=Obtained class, Cl=Hit or error in classification. For each class $p, n, tp, tn, fp,$ and $fn$ are obtained to get their accuracy, given in tables 9, 10, and 11 for equivalence and implication, operators and the deductive structure modus ponens.

Based on the results of classification with proposed symbolic predicates their accuracy was calculated (Tables 12, 13, and 14):

Comparing the accuracy of operators and deductive structure modus ponens proposed the equivalence obtained the highest (table 15):

For the experiment, 30 runs were executed and Table 16 shows a fragment of the comparisons of these runs between the operators of implication, equivalence and the deductive structure Modus Ponens. To compare the proposals and determine if one is significantly better than the others, the Wilcoxon test was used. For each pair of proposals, the following were compared:

1. the logical operators of implication and equivalence,

**Table 16.** Fragment of the Comparison of accuracy of operators and CDS

| Run | Imp | Eqv | MP |
|---|---|---|---|
| 1 | 76.5 | 93.6 | 87.9 |
| 2 | 77.3 | 90.7 | 83.6 |
| 3 | 78.7 | 90.7 | 87.6 |
| 4 | 84.9 | 90.8 | 83.1 |
| 5 | 77.8 | 91.4 | 85.1 |
| 6 | 77.3 | 94.3 | 90.4 |
| 7 | 75.6 | 95.6 | 81.8 |
| 8 | 76.4 | 87.8 | 82.1 |
| 9 | 76 | 97.8 | 83.1 |
| 10 | 77.8 | 94.8 | 78.7 |

**Table 17.** Bonferroni's test results

| G1 | G2 | stat | PV-Cor | Rej |
|---|---|---|---|---|
| EQV | IMP | 16.01 | 0 | True |
| EQV | MP | 10.21 | 0 | True |
| IMP | MP | -6.85 | 0 | True |

G1=Group1, G2=Group2,PV-Corr=pvalue-Corrected, Rej=rejected

**Table 18.** p-values

| | IMP | EQV |
|---|---|---|
| IMP | - | 1.86E-9 |
| EQV | 1.86E-9 | - |
| MP | 1.41E-6 | 1.86E-9 |

2. the equivalence operator with the modus ponens deductive structure and,
3. the equivalence operator with the deductive structure modus ponens.

The p-value is compared to 0.05, if it is lower, then there is a significant difference between the proposals and otherwise, there is no significant difference. Table 17 shows the p-values between the proposals.

All comparisons of the proposals obtained a p-value less than 0.05, so it is concluded that there is a significant difference between them.

Since there are significant differences between the proposals, the post hoc Bonferroni test is executed, obtaining the results shown in table 18: According to the results of the Bonferroni test, it is concluded that the significant difference is in favor of using the equivalence operator over the implication operator and the modus ponens deductive structure.

## 10 Conclusions

This work proved that a genetic algorithm that uses CFL provides a solid methodology that allows for solving classification problems through fuzzy predicates.

Moreover, the use of the equivalence (EQV) operator and the correct deductive structure modus ponens (MP) as a model compared with the implication operator contributes to a new level of classification.

The combination of a genetic algorithm that uses EQV or MP with classification models based on compensatory fuzzy logic generates predicates that tend to show better accuracy in the classification task.

Results of this comparison indicate that there is a tendency to obtain better accuracy using the equivalence operator with 94% over the implication with 76% and 88% of the predicate using the correct deductive structure.

Results are proof that this model allows to solve classification problems and these results serve in decision-making processes, data analysis, and data diagnosis, among others.

As future work, more experimentation remains to be done using other classification datasets as well as other methods of classification and the implementation of inference tasks.

## Acknowledgments

The authors want to thank Laboratorio Nacional de Tecnologías de Información and the support of (a) Cátedras CONAHCYT Program Number 3058, (b) the TecNM, (c) the support granted through the Scholarship for Postgraduate Studies with CVU 886000.

## References

1. **Cisneros, L., Rivera, G., Florencia, R., Sánchez-Solís, J. P. (2023).** Fuzzy optimisation for business analytics: A bibliometric analysis. Journal of Intelligent & Fuzzy Systems, Vol. 44, No. 2, pp. 2615–2630. DOI: 10.3233/JIFS-221573.
2. **Xexéo, G., Braga, A. (2008).** A tool for fuzzy reasoning and querying. Handbook of Research on Fuzzy Information Processing in Databases IGI Global. pp. 381–406. DOI: 10.4018/978-1-59904-853-6.ch015.
3. **Pérez-Pueyo, R. (2005).** Procesado y optimización de espectros Raman mediante técnicas de lógica difusa: aplicación a la identificación de materiales pictóricos. Universitat Politècnica de Catalunya. DOI: 10.5821/dissertation-2117-94198.
4. **Padrón-Tristán, J. F., Cruz-Reyes, L., Espín-Andrade, R. A., Llorente-Peralta, C. E. (2021).** A brief review of performance and interpretability in fuzzy inference systems. New Perspectives on Enterprise Decision-Making Applying Artificial Intelligence Techniques, pp. 237–266. DOI: 10.1007/978-3-030-71115-3_11.
5. **Zadeh, L. A. (1965)**. Information and control. Fuzzy sets, Vol. 8, No. 3, pp. 338–353. DOI: 10.1016/S0019-9958(65)90241-X.
6. **Serov, V. V., Sokolov, I. V., Budnik, A. A. (2019)**. Applied calculus of fuzzy predicates for the formalization of knowledge. In IOP Conference Series: Materials Science and Engineering, Vol. 537, No. 4, p. 042043. DOI: 10.1088/1757-899X/537/4/042043.
7. **Espin-Andrade, R. A., Gonzalez, E., Pedrycz, W., Fernandez, E. (2016)**. An interpretable logical theory: the case of compensatory fuzzy logic. International Journal of Computational Intelligence Systems, Vol. 9, No. 4, pp. 612–626. DOI: 10.1080/18756891.2016.1204111.
8. **Espín-Andrade, R. A., Bataller, A. C., Marx-Gomez, J., Racet, A. (2011).** Fuzzy semantic transdisciplinary knowledge discovery approach for business intelligence. Towards a Transdisciplinarian Technology for Business Intelligence: Gathering Knowledge Discovery, Knowledge Management and Decision Making, pp. 13–34.
9. **Galende, M., Sainz, G. I., Fuente, M. J. (2011).** Accuracy-interpretability trade-off for precise fuzzy modeling using simple indices. Application to Industrial plants. IFAC Proceedings, Vol. 44, No. 1, pp. 12656–12661. DOI: 10.3182/20110828-6-IT-1002.02005.
10. **Llorente-Peralta, C. E., Cruz-Reyes, L., Espín-Andrade, R. A. (2021).** Knowledge discovery using an evolutionary algorithm and compensatory fuzzy logic. Studies in Computational Intelligence, pp. 363-383. DOI: 10.1007/978-3-030-68776-2_21.

11. **González-Ramírez, C. M. (2017).** Aproximación al concepto de inferencia desde dos modelos de comprensión: modelo estratégico y modelo de construcción e integración. Literatura y lingüística, No. 35, pp. 295-312. DOI: 10.4067/S0716-58112017000100295.
12. **Bunge, M. (2000).** La investigación científica: su estrategia y su filosofía. Siglo XXI.
13. **Mamdani, E. H., Assilian, S. (1975).** An experiment in linguistic synthesis with a fuzzy logic controller. International Journal of Man-Machine Studies, Vol. 7, No. 1, pp. 1–13. DOI: 10.1016/S0020-7373(75)80002-2.
14. **Takagi, T., Sugeno, M. (1985).** Fuzzy identification of systems and its applications to modeling and control. IEEE transactions on systems, man, and cybernetics, No. 1, pp. 116–132. DOI: 10.1109/TSMC.1985.6313399.
15. **Galdi, P., Tagliaferri, R. (2018).** Data mining: accuracy and error measures for classification and prediction. Encyclopedia of bioinformatics and computational biology, Vol. 1, pp. 431–436. DOI: 10.1016/B978-0-12-809633-8-20474-3.
16. **Fernández-Cervantes, V., García, A., Ramos, M. A., Méndez, A. (2015).** Facial geometry identification through fuzzy patterns with RGBD sensor. Computación y Sistemas, Vol. 19, No. 3, pp. 529–546.
17. **Silva, D., Manzo-Martínez, A., Gaxiola, F., Gonzalez-Gurrola, L., Ramírez-Alonso, G. (2022).** Analysis of CNN architectures for human action recognition in video. Computación y Sistemas, Vol. 26, No. 2, pp. 623–641. DOI: 10.13053/cys-26-2-4245.
18. **Pach, F. P., Gyenesei, A., Abonyi, J. (2008).** Compact fuzzy association rule-based classifier. Expert Systems with Applications, Vol. 34, No. 4, pp. 2406–2416. DOI: 10.1016/j.eswa.2007.04.005.
19. **Binu, R., Isaac, P. (2021).** Multi-attribute neutrosophic decision-making in dosimetric assessment of radiotherapy imaging techniques. In: Smarandache, F., Abdel-Basset, M. (eds) Neutrosophic Operational Research. Springer, Cham. DOI: 10.1007/978-3-030-57197-9_28.
20. **Zadeh, L. A. (1975).** The concept of a linguistic variable and its application to approximate reasoning—I. Information sciences, Vol. 8, No. 3, pp. 199–249. DOI: 10.1016/0020-0255(75)90036-5.
21. **Gullo, F. (2015).** From patterns in data to knowledge discovery: What data mining can do. Physics Procedia, Vol. 62, pp. 18–22. DOI: 10.3109/0142159X.2014.1001729.
22. **Bigand, A., Colot, O. (2016).** Membership function construction for interval-valued fuzzy sets with application to Gaussian noise reduction. Fuzzy Sets and Systems, Vol. 286, pp. 66–85. DOI: 10.1016/j.fss.2015.07.001.
23. **Comas, D. S., Meschino, G. J., Nowé, A., Ballarin, V. L. (2017).** Discovering knowledge from data clustering using automatically-defined interval type-2 fuzzy predicates. Expert Systems with Applications, Vol. 68, pp. 136–150. DOI: 10.1016/j.eswa.2016.10.018.
24. **Zadeh, L. A. (1989).** Development of Probabilistic and Possebilistic Approaches to Approximate Reasoning and Its Applications. University of California, Computer Science Division, Berkeley. DOI: 10.21236/ADA221945.

*Article received on 29/02/2024; accepted on 15/05/2024.*
**Corresponding author is Carlos Eric Llorente Peralta.*

ISSN 2007-9737

# Deep Study on the Application of Machine Learning in Bin Packing Problems

Jessica González-San-Martín[1], Laura Cruz-Reyes[1,*], Bernabé Dorronsoro[2], Héctor Fraire-Huacuja[1], Fausto Balderas-Jaramillo[1], Marcela Quiroz-Castellanos[3], Nelson Rangel-Valdez[4]

[1] Tecnológico Nacional de México,
Instituto Tecnológico de Ciudad Madero,
Division of Graduate Studies and Research,
Mexico

[2] University of Cadiz,
Computer Science Engineering Cadiz,
Spain

[3] Universidad Veracruzana,
Artificial Intelligence Research Center,
Mexico

[4] CONAHCyT Research Fellow at Graduate Program Division,
Tecnológico Nacional de México,
Instituto Tecnológico de Ciudad Madero,
Mexico

jessica.gs@cdmadero.tecnm.mx

**Abstract.** This paper conducts a comprehensive review of literature focusing on strategies applied in the realm of Machine Learning (ML) in a period from ten years ago to the present to address the Bin Packing Problem (BPP) and its various variants. The Bin Packing Problem, a renowned optimization challenge, involves efficiently allocating items of varying sizes into containers of fixed capacity to minimize the number of containers used. Despite the extensive body of research and the existence of heuristic algorithms, unresolved challenges persist in BPP's solution. This deep study systematically explores the landscape of ML applications, delving into innovative approaches and methodologies proposed for tackling BPP and its diverse extensions, including 2D-BPP, 3D-BPP, Multi-objective BPP, and dynamic variants. The review critically examines the performance and contributions of ML-based strategies, shedding light on their efficacy in optimizing the packing process. Key findings highlight the promising directions taken by ML in solving complex optimization problems, emphasizing its potential to enhance BPP solution methodologies. The synthesis of diverse ML strategies and their integration with traditional heuristics forms a central theme, showcasing the evolving landscape of research in this domain. Additionally, this review identifies gaps and future research directions, emphasizing the relevance and effectiveness of ML as a valuable tool for improving performance in resolving BPP and its related challenges. The insights derived from this study aim to guide researchers, practitioners, and decision-makers in understanding the current state of ML applications in the context of Bin Packing Problems and inspire further advancements in this field.

**Keywords.** Bin packing problem, machine learning, metaheuristics, techniques and strategies.

## 1 Introduction

In recent years, the convergence of machine learning (ML) and metaheuristics has sparked great interest and innovation to address complex combinatorial optimization problems. Among these challenges, the Bin Packing Problem (BPP) and its

various extensions are presented as complex problems that demand sophisticated solutions.

This paper presents a comprehensive exploration of the synergies between ML and metaheuristics, focusing on their application to efficiently solve BPP and its various variants, such as 2D-BPP, 3D-BPP, multi-objective BPP, and dynamic BPP.

BPP involves strategically packing items of different sizes into fixed-size containers to minimize the number of used containers. Despite decades of research and the existence of numerous heuristic algorithms, unresolved challenges remain in achieving optimal solutions. In response to the limitations of traditional heuristic approaches, there has been growing interest in leveraging ML strategies to improve problem-solving capabilities.

This study delves into the novel realm of employing ML techniques, exploring various optimization methods and problem variants within the domain of Bin Packing. This study is carried out by incorporating works from ten years ago to the present to analyze the evolution of the applied ML strategies. Preliminary results underscore the promising trajectory of ML in addressing complex optimization problems, shedding light on its potential to revolutionize not only BPP resolution but also similar intricate challenges.

We examine the interaction between ML and metaheuristics, analyzing how ML contributes to problem modeling, decision-making, and adaptive search strategies to identify potential applications and chart a course for future research and developments in the pursuit of optimal solutions for container packing problems.

## 2 Bin Packing Problem

The Bin Packing Problem (BPP) is a classic challenge in combinatorial optimization. This problem is categorized as NP-hard due to its complexity [1]. In its most basic formulation, it is known as 1D-BPP (One-dimensional bin packing problem) and is presented with a set of elements of different sizes that must be assigned to containers of fixed capacity, seeking to minimize the total number of containers used. This problem is formally defined as follows:

Given a set $N = \{1, \dots, n\}$ of items to distribute in bins of the same size (capacity), let: $c =$ capacity of each bin and $w_i =$ weight of item $i$, such that $0 < w_i \leq c$ for $1 \leq i \leq n$.

This problem consists of assigning each item to a bin in such a way that the sum of the weights of the items in each bin does not exceed $c$ and the number of bins $m$ used is minimal [2]. We seek to find the least number of subsets $B_j$, for $1 \leq j \leq m$, of a partition of the set $N$:

$$\cup_{j=1}^{m} B_j = N. \tag{1}$$

Such as:

$$\sum_{\forall i \in B_j} w_i \leq c \quad 1 \leq j \leq m, i \in N = \{1, \dots, N\}. \tag{2}$$

Eq. (1) represents the number of bins that are used to pack the set of $N$ items, while Eq. (2) indicates that the sum of the weights of the items contained in the subset $B_j$ does not exceed the bin's capacity.

This problem, fundamental in logistics and planning, has generated an extensive field of research that covers various variants adapted to real-world situations [3]. Some of the variants of the BPP are defined below.

### 2.1 Problem Variants

#### Two-dimensional Bin Packing (2D-BPP)

In traditional BPP, elements are packed into a single dimension. However, in 2D-BPP are given a set of $n$ rectangular items $j \in J = \{1, \dots, n\}$, each characterized by its width $w_j$ and height $h_j$, alongside an unlimited supply of identical rectangular bins, each with fixed width $W$ and height $H$. The objective is to assign, without overlap, all items to the fewest number of bins possible, ensuring that their orientations align with those of the bins' edges. It is assumed that the items remain in a fixed orientation throughout the packing process, meaning they cannot be rotated.

Gilmore and Gomory [4] presented a mathematical formulation of 2D-BPP. They introduced a column generation strategy rooted in the exhaustive listing of all item subsets (known as patterns) that can be accommodated within a single bin. Here, let $A_j$ denote a binary column

vector comprising $n$ elements $a_{ij}$ (where $i = 1,\dots,n$), with a value of 1 assigned if item $i$ is part of the $j$th pattern, **and** 0 otherwise.

The set of all admissible patterns is thus denoted by the matrix $A$, which encompasses all conceivable $A_j$ columns (where $j = 1,\dots,M$) [5]. The associated mathematical formulation is expressed as:

$$min \sum_{j=1}^{M} x_j. \tag{3}$$

Such as:

$$\sum_{j=1}^{M} a_{ij}\, x_j = 1 \qquad (i = 1,\dots,n), \tag{4}$$

$$x_j \in \{0,1\} \qquad (j = 1,\dots,M). \tag{5}$$

This variant has many industrial applications, especially in cutting (e.g., wood, glass, and paper industries) and packing (e.g., transportation, telecommunications, and warehousing) [6].

### Three-dimensional Bin Packing (3D-BPP)

The 3D-BPP adds complexity by allowing the three-dimensional stacking of elements in three axes in bins. This results in greater flexibility in the arrangement of elements, but also poses new computational challenges when searching for the optimal three-dimensional configuration.

Martelo et al. [7] described the 3D-BPP as given a set of $n$ elements of rectangular shape, each characterized by width $w_j$, height $h_j$ and depth $d_j (j \in J = \{1,\dots,n\})$, and an unlimited number of identical three-dimensional bins having width $W$, height $H$, and depth $D$.The three-dimensional bin packing problem (3D-BPP) consists of orthogonally packing all items into the minimum number of containers.

Let $x_{ij}$ a binary variable that indicates whether element $j$ is placed in container $i$. It takes the value 1 if the element is placed in the bin and 0 otherwise (Eq. (11)):

$$min \sum_{i=1}^{M} \sum_{j=1}^{n} x_{ij}. \tag{6}$$

Such that:

$$\sum_{i=1}^{M} x_{ij} = 1 \qquad \forall j \in J, \tag{7}$$

$$\sum_{j=1}^{n} w_j \cdot x_{ij} \le W \qquad \forall i \in I, \tag{8}$$

$$\sum_{j=1}^{n} h_j \cdot x_{ij} \le H \qquad \forall i \in I, \tag{9}$$

$$\sum_{j=1}^{n} d_j \cdot x_{ij} \le D \qquad \forall i \in I, \tag{10}$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in I, \forall j \in J. \tag{11}$$

Eq. (7) indicates that each element must be assigned to exactly one bin. Eq. (8), (9), and (10) respectively indicate that the sum of the widths, heights, and depths of the elements placed in a container cannot exceed the width, height, and depth of the bin.

### Multi-objective Bin Packing Problem

The multi-objective variant of BPP (MOBPP) focuses on the simultaneous optimization of multiple criteria, such as minimizing the number of containers used, maximizing space utilization, and equitably distributing items among containers. This approach is especially relevant in situations where efficiency and performance objectives coexist and must be balanced.

Geiger [8] presented a problem formulation of MOBPP:

Given a number of $n$ items must be packed into $n$ bins, each of capacity $c$. Each item $j$ is characterized by a weight $w_j$ and an additional attribute $a_j$. While the weights refer to the size of the items and therefore must be taken into consideration when filling up a bin to at most its' capacity $c$, the attributes $a_j$ describe the properties of the items on a nominal scale. Based on this description, a comparison of two items $i, j$ is possible such that they are either identical concerning $a_i$ and $a_j$, $a_i = a_j$ or not: $a_i \neq a_j$. The goal of packing the items into bins can then be modeled as follows.

$$\min z_1 = \sum_{i=1}^{n} y_i, \tag{12}$$

$$\min z_2 = \frac{1}{z_1} \sum_{i=1}^{n} u_i. \tag{13}$$

Such as:

$$\sum_{j=1}^{n} w_j \cdot x_{ij} \leq c \cdot y_i \qquad i \in N = \{1, \ldots, n\}, \tag{14}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j \in N, \tag{15}$$

$$y_i = 0 \; or \; 1 \quad i \in N, \tag{16}$$

$$x_{ij} = 0 \; or \; 1 \quad i \in N, j \in N. \tag{17}$$

Eq. (12) minimizes the number of bins and the second objective given in Eq. (12) minimizes the average heterogeneousness of the bins.

**Dynamic Bin Packing Problem**

Unlike static versions, Dynamic BPP addresses problems where elements arrive sequentially, and the layout must be adjusted dynamically as new elements are introduced. Instead of knowing all the elements in advance, the challenge lies in making packaging decisions as the elements present themselves, requiring adaptive and efficient strategies. This scenario more accurately reflects real-time logistics situations and adds a layer of complexity to problem resolution.

Coffman et al. [9] described Dynamic Bin Packing as an extension of the classical static 1D-BPP model. The items to be packed will be described by a finite sequence or list $L\,(p_1, p_2, \ldots, p_n)$.

Each item $p_1$, in $L$ corresponds to a triple $(a_i, d_i, s_i)$, where $a_i$ is the arrival time for $p_i, d_i$ is its departure time, and $s_i$ is its size. The item $p_i$ resides in the packing for the time interval $[a_i, d_i)$, and we assume that $d_i - a_i > 0$ for all $i$. Without loss of generality, the common bin capacity will be taken always to be 1, so we also assume that each $s_i$ satisfies $0 < s_i \leq 1$ and that the items in $L$ are ordered so that $a_1 \leq a_2 \leq \cdots \leq a_n$.

Packaging rules denote that items cannot be moved from one bin to another once packed and operating online, i.e., items are packaged as they arrive without any knowledge of future arrivals. Therefore, the elements in $L$ bins will be allocated in order of increasing index, under the only restriction that at any time there is no container containing "currently active" elements whose sizes sum to more than 1.

This section provides a solid conceptual framework for understanding BPP and its various variants, laying the foundation for exploring how Machine Learning strategies address these challenges in the next section.

## 3 Machine Learning in Metaheuristics

The employment of Machine Learning (ML) in combination with metaheuristics has emerged as an innovative and effective approach to address combinatorial optimization problems, including the challenging BPP and its variants. This section explores the convergence of ML and metaheuristics, presenting a detailed view of their integration and application in efficiently solving these problems.

Machine Learning focuses on the development of algorithms and models capable of learning patterns from data and dynamically adapting to new situations. This adaptive capacity is essential to address combinatorial optimization problems, where effective solutions must be discovered through the exploration and exploitation of multiple options.

The synergy between ML and metaheuristics lies in the ability of ML models to improve metaheuristic adaptive capabilities. By incorporating machine learning techniques, metaheuristics can learn and adjust their search strategies adaptively as they interact with the solution space.

Talbi [10] presents a general taxonomy of how metaheuristics incorporate machine learning. Fig. 1., presents three ways of applying machine learning to a metaheuristic:

– **Problem-level Machine Learning support in metaheuristics**: Machine Learning can play a crucial role in modeling optimization problems, addressing aspects such as the objective

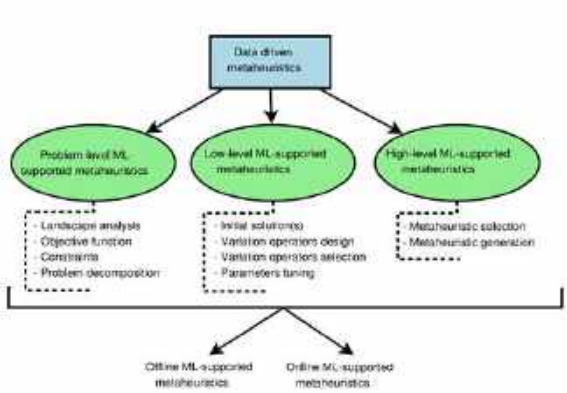



**Fig. 1.** Talbi´s general taxonomy of machine learning-supported metaheuristics [10]

function and constraints. In addition, it can contribute to the analysis of the problem landscape and its decomposition.

– **Metaheuristics with low-level ML support:** Metaheuristics are made up of various search components. In this context, Machine Learning can drive any search component, from solution initialization to search variation operators such as neighborhoods in local search, as well as mutation and crossover in evolutionary algorithms. Likewise, it can be used to adjust the different parameters of a metaheuristic.

– **High-level ML-supported metaheuristics**: This category focuses on the selection and generation of metaheuristics, as well as the design of hybrid strategies and cooperative metaheuristics in parallel.

At the same time, other criteria related to learning time are used. For offline ML-supported metaheuristics, the learning process is performed before the execution of the problem solver, this allows the model to be trained in advance.

On the other hand, in online ML-supported metaheuristics, the machine learning process is carried out concurrently with the search and resolution of the problem, allowing the model to accumulate knowledge in real time during the optimization process.

Among the machine learning techniques most used in metaheuristics, some strategies stand out:

– **Supervised Learning:** Uses labeled datasets to train models and guide adaptive capabilities in the metaheuristic.
*Applications*: Used to model the objective function, constraints, and other aspects of the optimization problem.
*Benefits*: Enhances solution quality by incorporating learned patterns from previous examples.

– **Unsupervised Learning:** Explores patterns in unlabeled data, allowing the metaheuristic to discover hidden structures and adapt to problem complexity.
*Applications*: Useful for analyzing solution spaces and dynamically adjusting the search.
*Benefits*: Provides flexibility and adaptability to changing conditions.

– **Reinforcement Learning:** Involves the dynamic interaction of the agent (metaheuristic) with the environment (optimization problem), aiming to maximize rewards over time.
*Applications*: Guides exploration and exploitation of solutions, optimizing the metaheuristic's performance.
*Benefits*: Adapts search strategies based on accumulated experiences, improving efficiency over time.

– **Deep Learning:** Utilizes deep neural networks to learn complex representations and make predictions.
*Applications*: Applicable in high-dimensional optimization problems, allowing the capture of more abstract patterns.
*Benefits*: Improves modeling and generalization capabilities, especially in complex and nonlinear problems.

– **Hyperparameter Optimization:** Uses ML techniques to dynamically adjust metaheuristic parameters during the search.
*Applications*: Optimizes the metaheuristic configuration to enhance performance across different problems.
*Benefits*: Improves adaptability and efficiency of the metaheuristic in diverse scenarios.

– **Artificial Neural Networks (ANN):** Computational models inspired by the

structure and function of the human brain, capable of learning complex patterns.

*Applications*: Used to approximate objective functions, represent problem landscapes, and adapt in real-time during the search.

## 4 Hybrid Methods in BPP

Research aimed at addressing the challenges inherent in BPP has seen notable advances in recent years, especially with the incorporation of innovative strategies and advanced machine learning techniques. Over the years, various studies have significantly delved into these combinatorial optimization problems.

In this section, a summary of the literature that has used various ML techniques to address BPP from ten years ago to the present is presented. The twenty selected works have been organized according to the different techniques used by the authors, thus providing a detailed overview of the diversity of approaches applied in research on this topic.

### 4.1 Neural Networks

De Almeida and Steiner [11] address the 1D-BPP, exploring different configurations of AugNN. The study provides a comparative analysis between AugNN and Minimum Bin Slack (MBS), offering insights through experimental design and statistical analysis.

Sim et al. [12] present a hyperheuristic system that continuously learns over time to solve the 1D-BPP. The system continually generates new heuristics and shows problems in its environment; Representative problems and heuristics are incorporated into a self-sustaining network of interacting entities inspired by artificial immune system methods.

The network is plastic in both its structure and content, leading to the following properties: it exploits existing knowledge captured in the network to rapidly produce solutions; can adapt to new problems with very different characteristics; and is able to generalize over the problem space.

Laterre et al. [13] developed a ranked reward algorithm (R2) utilizing deep neural networks for solving the 2D and 3D Bin Packing Problems. The algorithm estimates a policy and a value function using deep neural networks, combined with Monte Carlo Tree Search (MCTS) for policy improvement.

Kroes et al. [14] utilize conventional neural networks to tackle the 1D-BPP. The study explores the hybridization of genetic algorithms and simulated annealing with traditional neural network heuristics for flexible container memory mapping.

### 4.2 Deep and Reinforcement Learning

Hu et al. [15] presented a new type of 3D-BPP, where a series of cuboid-shaped elements must be placed in an orthogonal-shaped container, one by one. The goal is to minimize the surface area of the container, reflecting real business scenarios where the cost of the container is proportional to its surface area. An approach based on Deep Reinforcement Learning (DRL), especially the Pointer Network, is applied to optimize the sequence of elements to be packed.

Mao et al. [16] introduce a deep learning approach to the Variable-Sized 1D Bin Packing Problem (1D-VSBPP), using a large training dataset and techniques for automatic feature selection and rapid labeling. They demonstrate how to build an adaptive system that can select the best heuristics to generate high-quality container packaging solutions.

Nanda and Hacker [17] employed deep reinforcement learning to consolidate active containers with different resource requirements into a minimal number of physical machines, proposing the RACC (Resource-Aware Container Consolidation) algorithm. Considers the resource demands of different types of tasks and the heterogeneity of physical machines.

Li and Hu [18] developed a deep reinforcement learning-based job scheduling algorithm for the container packing problem in cloud data centers. DeepJS can automatically obtain a fitness calculation method which will minimize the makespan (maximize the throughput) of a set of jobs directly from experience and the results prove that DeepJS outperforms the heuristic-based job scheduling algorithms.

Verma et al. [19] introduce a deep reinforcement learning algorithm for the online 3D Bin Packing Problem, focusing on decisions that

can be physically implemented using a robotic loading arm.

This approach tackles a novel problem in two key aspects: Firstly, it assumes an unknown set of objects to be packed, with only a fixed number visible to the loading system upon arrival. Secondly, the objective is not merely to move objects but to optimize their location and orientation within the bin(s) to maximize packing efficiency.

Goyal and Deng [20] present PackIt, a virtual environment for evaluating and potentially learning the ability to perform geometric planning. It models the 3D-BPP, where an agent needs to sequence actions to pack a set of objects into a box with limited space. They construct challenging packing tasks using an evolutionary algorithm, including model-free and heuristic-based learning methods, as well as optimization methods based on searches assuming access to the environment model.

Silva-Gálvez et al. [21] introduced a solution model for the 1D-BPP that leverages unsupervised learning principles within a hyper-heuristic framework. The model dynamically selects from various heuristics during the search process, adapting to the problem state being explored.

By employing the k-means clustering algorithm, they identify action regions and recommend heuristics based on performance analysis within these regions. The hyper-heuristic determines the most appropriate heuristic to apply based on the current problem state.

Zhao et al. [22] present a restricted deep reinforcement learning method for the 3D Bin Packing Problem, using a feasibility predictor to modulate the output probabilities during training.

They introduced a prediction-and-projection scheme: The agent first predicts a feasibility mask for the placement actions as an auxiliary task and then uses the mask to modulate the action probabilities output by the actor during training. Such supervision and projection facilitate the agent to learn feasible policies very efficiently. The method can be easily extended to handle lookahead items, multi-bin packing, and item re- orienting.

Yang et al. [23] developed a flexible container packing framework based on slack, using reinforcement learning strategies to generate slack and improve container space efficiency.

The performance-driven rewards are used to capture the intuition of learning the current state of the container space, the action is the choice of the packing container, and the state is the remaining capacity after packing. During the construction of the slack, an instance-eigenvalue mapping process is designed and utilized to generate the representative and classified validate set

Zhang et al. [24] present an end-to-end learning model for bin packing using self-attention and deep reinforcement learning algorithms. They introduce the prioritized oversampling technique to accelerate policy learning. Their proposal is called Att2pack, which was initially proposed to address the offline 1D-BPP, however, they adapted it to the online BPP which also shows good performance.

Jiang and Zhang [25] develop an end-to-end multimodal deep reinforcement learning agent to solve the 3D-BPP, addressing sequence, orientation, and position tasks. They use a multimodal encoder and maintain lightweight computation to learn the packing policy.

Murdivien and Um [26] This study investigates the use of Deep Reinforcement Learning (DRL) to address dynamic logistics challenges, focusing on the real-time 3D sequential packing problem. It highlights the need for adaptive and autonomous manufacturing systems in the face of sudden changes in the supply chain. By employing neural networks in reinforcement learning, DRL shows promise in solving complex problems in manufacturing logistics. A game engine is used to train the DRL, allowing intuitive visualization of the learning process.

Fang et al. [27] introduce a reinforcement learning algorithm based on Monte Carlo (MC) learning, Q-learning, and Sarsa-learning to address a 2D irregular packing problem. The proposed algorithm adapts reward-return mechanisms and updates strategies based on the irregular piece-packing scenario.

Guerriero and Saccomanno [28] tackle the 1D-BPP using a reinforcement learning strategy aimed at training an agent to mimic a reference heuristic. In particular, the reward is proportional to the agent's ability to make the same decisions as a particular heuristic when applied to a specific problem state.

### 4.3 Others

Duan et al. [29] propose a multitask learning framework based on reinforcement learning for the Flexible 3D Bin Packing Problem (3D-FBPP). It focused on minimizing the plastic surface area used to wrap packages, which involves packing cuboid-shaped items in bins with the smallest possible surface area.

Instead of designing heuristics, they proposed a multi-task learning framework based on Selected Learning to learn a heuristic-like policy that generates the sequence and orientations of the elements to be packed simultaneously.

Mohiuddin et al. [30] focused on the challenges associated with cloud storage, such as inefficient use of resources and internal threats to stored data.

It proposes a distributed storage allocation architecture to ensure equitable use of storage resources and an integrated security framework to protect data at rest in cloud storage from insider threats. The goal is to effectively address challenges related to resource management and data security in the cloud.

The analysis of the works highlights the effectiveness of strategies based on neural networks, deep learning algorithms, and reinforcement learning methods to enhance the efficiency and quality of solutions in the field of container packaging.

However, it is crucial to note that the dynamic variants of this problem have received limited exploration so far. This research gap could represent a valuable area of opportunity for future studies, as the dynamics of ever-changing environments present additional challenges that could benefit from innovative approaches.

Additionally, it is worth noting that reinforcement learning (RL) and deep reinforcement learning (DRL) have emerged as preeminent strategies to address the challenges of container packaging, being recurrently used in the majority of the studies reviewed.

The combination of these approaches with other methodologies has been shown to offer promising results, underlining the versatility and effectiveness of these techniques in this domain.

## 5 Relevant State-of-the-Art Methods

The previous section provided an overview of work related to the BPP, exploring machine learning strategies and highlighting techniques employed by various researchers.

Now, we will immerse ourselves in a critical analysis, focusing on the selection of three representative works considering the number of citations of each one in the reviewed literature. These studies play a crucial role in the evolution of methods and approaches to optimize BPP.

We will examine the techniques and algorithms implemented in these works. This analysis will not only highlight the strengths and limitations of each approach but will also establish a solid framework for understanding the broader landscape of research in this field.

We also perform an in-depth examination of the practical implications and results obtained by the three selected methodologies for solving Bin Packing Problems. Table 1 presents the general characteristics of the three works selected as a case study.

The name of the algorithm and author, year of publication, number of citations, BPP variant that is addressed, and the main ML strategies that apply are included.

### 5.1 A Lifelong Learning Hyper-heuristic Method for Bin Packing

Sim et al. [12] introduced a novel hyperheuristic approach designed to continuously learn and tackle combinatorial optimization problems. This system dynamically generates fresh heuristics when confronted with challenges in its environment.

Representative problems and heuristics become integral parts of an autonomous network of interacting entities, drawing inspiration from artificial immune system methods.

The flexibility in both structure and content within the network imparts noteworthy qualities: leveraging existing knowledge for swift solution generation, effective adaptation to new problems with diverse characteristics, and the ability to generalize within the problem space.

The system's efficacy is evaluated using an extensive dataset of one-dimensional bin packing

**Table 1.** General characteristics of works representative of the state-of-the-art

| Algorithm | Year | Citations | BPP variant | Strategies / methods |
|---|---|---|---|---|
| NELLI Sim et al. [11] | 2015 | 90 | 1D-BPP | Hyper-heuristic system (artificial immune system methods and neuronal network). |
| Packman Verma et al. [18] | 2020 | 40 | 3D-BPP | Deep reinforcement learning algorithm with a focus on robotic loading arm implementation. |
| RS Zhao et al., [21] | 2021 | 107 | 3D-BPP | Deep reinforcement learning with a feasibility predictor scheme. |

problems (1D-BPP) and 1370 problems previously established in the literature. The results showcase exceptional performance in terms of solution quality across all datasets.

The system exhibits efficient adaptation to sets of problem instances undergoing dynamic changes, surpassing the performance of previous approaches.

The network's capacity to self-adjust and maintain a concise repertoire of problems and heuristics, functioning as a representative map of the problem space, underscores the computational efficiency and scalability of the system.

Algorithm 1 presents the pseudocode of the work (NELLI) proposed by Sim et al.

**Algorithm 1:** NELLI pseudocode

**Require:** $\mathcal{H} = \emptyset$: The set of heuristics
**Require:** $\mathcal{P} = \emptyset$: The set of current problems
**Require:** $\mathcal{E} = \mathcal{E}_{t=0}$: The set of problems to be solved at time $t$

1: **repeat**
2: *Optionally* replace $\mathcal{E} : \mathcal{E}^* \leftarrow \mathcal{E}^* \cup \mathcal{E}$
3: Add $n_h$ randomly generated heuristics to $\mathcal{H}$ with concentration $c_{init}$
4: Add $n_p$ randomly selected problem instances from $\mathcal{E}$ to $\mathcal{P}$ with *concentration* $c_{init}$
5: Calculate $h_{stim} \forall h \in \mathcal{H}$
6: Calculate $p_{stim} \forall p \in \mathcal{P}$
7: Increment all concentrations (both $\mathcal{H}$ and $\mathcal{P}$) that have *concentration* $< c_{max}$ and stimulation $> 0$ by $\Delta_c$
8: Decrement all concentrations (both $\mathcal{H}$ and $\mathcal{P}$) with stimulation $< 0$ by $\Delta_c$
9: Remove heuristics and problems with *concentrations* $\leq 0$
10: **until** *stopping criteria met*

**Algorithm 1.** The pseudocode of the NELLI algorithm presented by Sim et al. [12]

The results reported by Sim et al. [12] include a comparison of the performance of the proposed algorithm (NELLI) against literature works previously developed by one of the collaborators: AIS I [31] and Island [32].

They used different instances to evaluate the Bin Packing Problem (BPP) such that Data Set 1, 2, 3 [33], Uniform, and Triplets [34].

Table 2 presents the results obtained by NELLI, AIS I, and Island for a set of 685 instances tested by the authors.

Table 3 presents the evaluation of NELLI vs Island for 1370 instances, taking into account the minimum, maximum, mean, and standard deviation of each of the algorithms for both comparisons.

According to the analysis of Sim et al. [12] Tables 2 and 3 presents that both NELLI and Island systems generate solutions of identical quality on a data set of 685 and 1370 instances.

However, it is important to highlight that NELLI has several advantages compared to the evaluated approaches. Its scalability is greater since it significantly reduces the calculation time compared to the evaluated approaches.

Furthermore, NELLI has been shown to adapt efficiently to non-visible problems and dynamic environments, maintaining a memory of previous experiences.

**Table 2.** NELLI results vs related works for a set of 685 instances [12]

| Algorithms | Problems solved | | | |
|---|---|---|---|---|
| | **Min** | **Max** | **Mean** | **SD** |
| AIS I [30] | 554 | 559 | 556 | 1.4 |
| Island [31] | 552 | 559 | 557 | 1.4 |
| NELLI [11] | 559 | 559 | 559 | 0 |

**Table 3.** NELLI [12] results vs Island [32] for a set of 1370 instances

| Algorithms | Problems solved | | | |
|---|---|---|---|---|
| | **Min** | **Max** | **Mean** | **SD** |
| Island [31] | 1120 | 1126 | 1125 | 1.1 |
| NELLI [11] | 1125 | 1126 | 1126 | 0.3 |

### 5.2 A Generalized Reinforcement Learning Algorithm for Online 3D Bin-Packing

Verma et al. [18] introduced a Deep Reinforcement Learning (Deep RL) algorithm to tackle the challenge of online 3D bin packing (3D-BPP), accommodating a variable number and any container size. The approach focuses on decision-making that can be physically implemented through a robotic loading arm, validated using a laboratory prototype.

This problem presents innovation in two key aspects. Firstly, in contrast to the conventional 3D-BPP, it assumes that the complete set of objects to be packed is not known in advance. Instead, the loading system observes a fixed number of nearby objects and must load them in the order they arrive.

Secondly, the goal is not merely to move objects from one point to another through a feasible route, but to find the location and orientation for each object that maximizes the overall packing efficiency of the containers. Additionally, the learned model is designed to handle instances of problems of arbitrary size without the need for retraining.

Simulation results indicate that the RL-based method outperforms state-of-the-art heuristics for online container packing, demonstrating improvements in terms of empirical competitive ratio and volume efficiency. The authors do not present the pseudocode of the proposed algorithm; however, they show the network architecture of the DQN agent used (Fig. 2).

Verma et al. [19] propose the PackMan algorithm, which was trained using synthetically generated data sets, which contain randomly generated dimension boxes. However, they made sure that the dimensions matched so that each container could be filled (100% fill fraction).

Each data set consists of 10 box bins ($Opt(I) = 10$), and the number of boxes ranges from 230 to 370 per episode. The initial baling efficiency of about 65% steadily improves to 82% in 1100 episodes and remains stable thereafter. The number of containers used decreased from just over 16 to just under 13.

Table 4 compares the algorithms on the competitiveness ratiometric ($c$), the time taken per loading decision, average packing efficiency, and the fraction of test instances in which a given algorithm returned the best packing efficiency. Advanced Harmonic (AH) is known to have a theoretical upper bound of $c$ = 1.58, although this is with unconstrained rotation.

Empirical results for robot stackable algorithms show that PackMan has the best empirical ratio of 1.29, averaging $T_{used}$ = 12.9 bins compared to $Opt(I)$ = 10. It also has the highest average packing fraction. While the difference in packing fractions is small, further investigation revealed that this was because there was significant variation among the instances, with some box streams favoring one algorithm over the others. The fact that PackMan returns the best efficiency in 57% of test cases implies that it retains a significant advantage over other algorithms across a variety of instances [19].

The boxplot shown in Fig. 3. illustrates the differences between the algorithms. While floor and column construction have almost identical results for the test data sets, WallE returns the best results among the heuristics.

This is a result of their balanced approach to box placement, with no singular emphasis on floor or column construction. The average packing efficiency for PackMan is higher than all heuristics but has a larger spread in outliers.

### 5.3 Online 3D Bin Packing with Constrained Deep Reinforcement Learning

Zhao et al. [22] address a variant of the challenging but highly practical 3D Container Packing Problem (3D-BPP). In the proposed scenario, the agent has limited information about the items that need to be packed into a single container, and each item must be placed immediately with no option to readjust later. The arrangement of items is also influenced by constraints related to order dependency and physical stability.

To address this online 3D-BPP, they propose a strategy based on the Constrained Markov Decision Process (CMDP). To solve this problem, they propose an efficient and easy-to-implement method based on constrained deep reinforcement learning (DRL), within the actor-critic framework. Specifically, they introduce a prediction and projection scheme: the agent initially anticipates a feasibility mask for location actions as an additional task and then uses this mask to adjust the action probabilities generated by the actor during training.

This monitoring and projection approach make it easier for the agent to learn viable policies in a highly efficient manner. The results of a comprehensive evaluation demonstrate that the learned policy significantly outperforms state-of-the-art methods in this context.

To validate their work called RS, Zhao et al. [22] carried out comparisons with two different methods. First, they evaluated their online approach against the BPH heuristic [35], which allows the agent to select the next best element among $k$ anticipated elements (i.e., BPP-$k$ with reordering).

In Table 5, it was specifically compared to the BPP-1 version of BPH. Also, in Fig. 4, the online BPH and the method under the BPP-$k$ configuration was compared. Second, they evaluated their method against the offline LBP method [7], which incorporates a baseline heuristic called the bounds rule method.

The latter imitates human behavior by trying to place a new item next to already packaged items, seeking to keep the packaging volume uniform. The comparison in Figure 8 reveals that the proposed method outperforms all online methods on all three benchmarks and surprisingly even outperforms the offline approach on CUT-1 and CUT-2.

Table 5 presents the method proposed by Zhao et al. [22] compared against three other approaches both online and offline, the number of objects and the percentage of space used by each approach are measured.

From the comparison in Table 5, the proposed method outperforms all alternative online methods on all three benchmarks and even beats the offline approach on CUT-1 and CUT-2.

Through examining the packing results visually, we find that our method automatically learns the above “boundary rule” even without imposing such constraints explicitly.

From Fig. 4, the method performs better than online BPH consistently with varying numbers of lookahead items even though BPH allows re-ordering of the lookahead items. They also

conducted a preliminary comparison on a real robot test of BPP-1 (see our accompanying video). Over 50 random item sequences, our method achieves an average of 66.3% space utilization, much higher than the boundary rule (39.2%) and online BPH (43.2%).

The reviewed works have provided valuable insights into the realm of BPP, shedding light on machine learning strategies and diverse methodologies employed by researchers. The critical analysis of three representative works has contributed to a deeper understanding of the field's evolution and progress.

Sim et al. [12] approach introduces a groundbreaking hyperheuristic method, showcasing adaptability, swift problem-solving capabilities, and the ability to generalize within the problem space. The system's commendable performance across extensive datasets and dynamic problem instances underscores its computational efficiency and scalability.

Verma et al. [19] Deep RL algorithm for online 3D bin packing presents innovation by addressing unknown object sets and optimizing container packing efficiency. The model's ability to handle instances of varying sizes without retraining is a notable strength, outperforming existing heuristics in terms of competitive ratio and volume efficiency.

Zhao et al. [22] strategy for the 3D Container Packing Problem stands out in its approach to immediate item placement, constrained by limited information and influenced by order dependency and stability considerations. The proposed CMDP-based method, utilizing constrained DRL, exhibits remarkable efficiency in learning viable policies, surpassing state-of-the-art methods in the context of online 3D-BPP.

When comparing the three approaches in the literature, different aspects and performances stand out. NELLI [12] has demonstrated notable advantages in terms of scalability and adaptability to dynamic environments, outperforming other previous approaches. For its part, the PackMan algorithm [19] presents significant improvements in packaging efficiency, achieving more efficient space utilization.

However, it suffers from greater variability in results. Zhao et al. [22], with its reinforcement learning-based approach, exhibit outstanding superiority over online and offline methods,

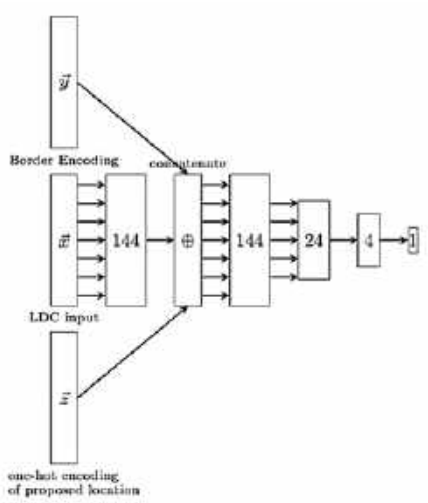



**Fig. 2.** Network architecture for the DQN agent presented in Verma et al. [18]

**Table 4.** Comparison of results on 100 episodes of test data between Packman vs other approaches from the literature [19]

| Algorithm | Comp. ratio $c$ | Time per box (sec) | Avg. pack | Best Pack |
|---|---|---|---|---|
| AH | 1.58 | - | - | - |
| Floor building | 1.52 | 0.0002 | 81.0% | 5% |
| Column build | 1.46 | 0.0001 | 81.0% | 6% |
| First Fit | 1.47 | 0.0002 | 81.3% | 7% |
| WallE | 1.41 | 0.0106 | 81.8% | 25% |
| PackMan | **1.29** | 0.0342 | **82.8%** | **57%** |

achieving efficient space utilization compared to traditional heuristics and even outperforming human players in certain scenarios.

Together, these results underscore the diversity of approaches and particular strengths of each algorithm. Considering the specific characteristics of each problem and the requirements of the environment, the choice of the most appropriate approach will depend on the specific objectives and constraints of the application.

In essence, these representative works contribute significantly to the ongoing

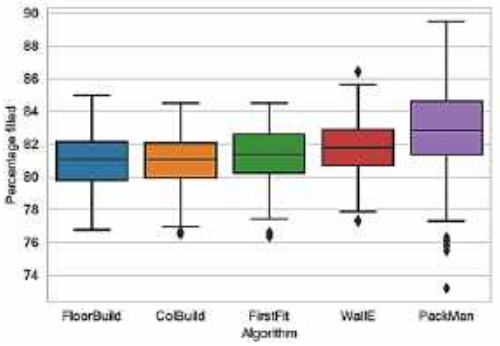



**Fig. 3.** Comparison of empirical compliance rates for the 5 algorithms, over 100 test data sets presented in [19]

advancements in BPP research, offering novel perspectives, innovative methodologies, and promising results.

The field continues to evolve, and these studies serve as crucial milestones in shaping the landscape of optimization strategies for Bin Packing Problems.

## 6 Conclusions and Future Directions

### 6.1 Conclusions

Despite promising advances in hybridizing machine learning strategies with the container packaging problem (BPP) and its variants, there are untapped areas of opportunity that deserve further exploration. Dynamic variants of BPPs emerge as relatively unexplored territory within this context, providing substantial space for future research and the development of novel hybrid strategies.

Furthermore, while deep reinforcement learning is an effective tool in most of the studies reviewed, there is a pressing need to deepen our understanding of their applicability in various BPP scenarios. Opportunities abound to investigate and improve learning adaptability to changing conditions, as well as explore its integration with other machine learning techniques and metaheuristics.

These areas of opportunity suggest that, despite commendable achievements to date, the trajectory of machine learning with BPP remains in a state of continuous evolution. The landscape continues to offer fertile ground for future research, driving continued refinement of the effectiveness and versatility of hybrid strategies.

### 6.2 Future Directions

The research landscape in the field of Bin Packing Problems (BPP) continues to evolve, driven by advancements in algorithms and methodologies. Building upon the insights gained from the analysis of the selected studies, several promising avenues for future research emerge.

1 Dynamic and Adaptive Algorithms: While some studies have touched upon dynamic aspects of BPP, there remains ample room for the development of algorithms that can dynamically adapt to changing scenarios. Future research could focus on the design of algorithms capable of adjusting their strategies in real-time as the packing environment evolves.
2 Integration of Hybrid Approaches: Combining the strengths of different algorithms and techniques can potentially lead to more robust and efficient solutions. Future research might explore the integration of machine learning, metaheuristics, and mathematical programming to create hybrid approaches that leverage the complementary advantages of these methods.
3 Scalability and Generalization: Many existing algorithms excel in specific scenarios but struggle with scalability or fail to generalize across diverse problem instances. Future work could concentrate on enhancing the scalability of algorithms, enabling them to handle larger problem sizes, and improving their generalization capabilities to address a broader range of BPP variants.
4 Real-world Implementation and Validation: As algorithms mature, practical implementation in real-world settings becomes crucial. Future research should aim at validating proposed algorithms through deployment in industrial or logistical environments, assessing their performance under actual operational conditions.
5 Incorporating Environmental Considerations: With a growing emphasis on sustainability, there is an opportunity to integrate

**Table 5.** Comparison of Zhao's proposal [22] against three online and offline approaches

| Method | # items / % space uti. | | |
|---|---|---|---|
| | **RS** | **CUT-I** | **CUT-2** |
| Boundary rule (Online) | 8.7 / 34.9% | 10.8 / 41.2% | 11.1 / 40.8% |
| BPH (Online) | 8.7 / 35.4% | 13.5 / 51.9% | 13.1 / 49.2% |
| LBP (Offline) | 12.9 / 54.7% | 14.9 / 59.1% | 15.2 / 59.5% |
| Proposed BPP-1 (Online) | 12.2 / 50.5% | 19.1 / 73.4% | 17.5 / 66.9% |

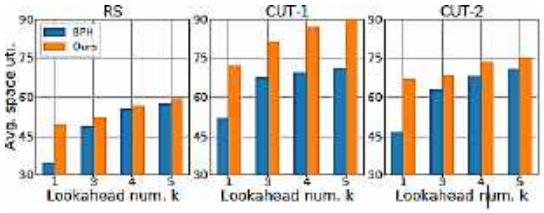



**Fig. 4.** Comparison with the online BPH method [28] in BPP-$k$. BPH allows for the reordering of anticipated items, while the work of [18] does not

environmental considerations into BPP algorithms. Future research might explore how to optimize packing solutions with a focus on reducing ecological impact, considering factors such as fuel efficiency in transportation or eco-friendly packaging materials.

6 User Interaction and Explainability: Developing algorithms that are interpretable and allow for user interaction can enhance their practical utility. Future research could delve into incorporating explainable AI principles and user-friendly interfaces, enabling stakeholders to understand and interact with the decision-making process.

7 Benchmark Datasets and Competitions: Establishing standardized benchmark datasets and organizing competitions can foster collaborative advancements in the field. Future efforts could focus on creating benchmark datasets that capture the complexities of real-world packing scenarios and organizing competitions to benchmark the performance of different algorithms on these datasets.

In pursuing these future directions, researchers can contribute to the continued evolution of BPP methodologies, addressing emerging challenges and ensuring the practical relevance of their findings.

## Acknowledgments

The authors want to thank Laboratorio Nacional de Tecnologías de la Información and the support of (a) Cátedras CONAHCYT Program Number 3058, (b) the TecNM, (c) the support granted through the Scholarship for Postgraduate Studies with CVU 960518.

## References

1. **Falkenauer, E. (1996).** A hybrid grouping genetic algorithm for bin packing. Journal of Heuristics, Vol. 2, pp. 5–30. DOI: 10.1007/BF00226291.
2. **Martello, S., Toth, P. (1990).** Lower bounds and reduction procedures for the bin packing problem. Discrete applied mathematics, Vol. 28, No. 1, pp. 59–70. DOI: 10.1016/0166-218X(90)90094-S.
3. **Rivera, G., Cisneros, L., Sánchez-Solís, P., Rangel-Valdez, N., Rodas-Osollo, J. (2020).** Genetic algorithm for scheduling optimization considering heterogeneous containers: A real-world case study. Axioms, Vol. 9, No. 1, pp. 27. DOI: 10.3390/axioms9010027.
4. **Gilmore, P. C., Gomory, R. E. (1965).** Multistage cutting stock problems of two and more dimensions. Operations research, Vol. 13, No. 1, pp. 94–120. DOI: 10.1287/opre.13.1.94.
5. **Lodi, A., Martello, S., Monaci, M. (2002).** Two-dimensional packing problems: A survey. European journal of operational research, Vol. 14, No. 2, pp. 241–252. DOI: 10.1016/s0377-2217(02)00123-6.

6. **Lodi, A., Martello, S., Monaci, M., Vigo, M. (2014).** Two-dimensional bin packing problems. Paradigms of combinatorial optimization: Problems and new approaches, In Paschos, V. T. (eds), pp. 107–129. DOI: 10.1002/9781119005353.ch5.
7. **Martello, S., Pisinger, D., Vigo, D. (2000).** The three-dimensional bin packing problem. Operations research, Vol. 48, No. 2, pp. 256–267. DOI: 10.1287/opre.48.2.256.12386
8. **Geiger, M. J. (2008).** Bin packing under multiple objectives heuristic approximation approach. arXiv:0809.0755. DOI: 10.48550/arXiv.0809.0755.
9. **Coffman-Jr, E. G., Garey, M. R., Johnson, D. S. (1983).** Dynamic bin packing. SIAM Journal on Computing, Vol. 12, No. 2, pp. 227–258. DOI: 10.1137/0212014.
10. **Talbi, E. G. (2021).** Machine learning into metaheuristics: A survey and taxonomy. ACM Computing Surveys (CSUR), Vol. 54, No. 6, pp. 1–32. DOI: 10.1145/3459664.
11. **De-Almeida, R., Steiner, M. T. A. (2015).** Resolution of 1-D bin packing problem using augmented neural networks and minimum bin slack. 2015 Latin America Congress on Computational Intelligence, IEEE, pp. 1–6. DOI: 10.1109/LA-CCI.2015.7435943.
12. **Sim, K., Hart, E., Paechter, B. (2015).** A lifelong learning hyper-heuristic method for bin packing. Evolutionary computation, Vol. 23, No. 1, pp. 37–67. DOI: 10.1162/EVCO_a_00121.
13. **Laterre, A., Fu, Y., Jabri, M. K., Cohen, A. S., Kas, D., Hajjar, K., Beguir, K. (2019).** Ranked reward: enabling self-play reinforcement learning for bin packing.
14. **Kroes, M., Petrica, L., Cotofana, S., Blott, M. (2020).** Evolutionary bin packing for memory-efficient dataflow inference acceleration on FPGA. Proceedings of the 2020 Genetic and Evolutionary Computation Conference, pp. 1125–1133. DOI: 10.1145/3377930.3389808.
15. **Hu, H., Zhang, X., Yan, X., Wang, L., Xu, Y. (2017).** Solving a new 3d bin packing problem with deep reinforcement learning method. arXiv:1708.05930. DOI: 10.48550/arXiv.1708.05930.
16. **Mao, F., Blanco, E., Fu, M., Jain, R., Gupta, A., Mancel, S., Tian, Y. (2017).** Small boxes big data: A deep learning approach to optimize variable sized bin packing. 2017 IEEE Third International Conference on Big Data Computing Service and Applications, IEEE, pp. 80–89. DOI: 10.1109/BigDataService.2017.18.
17. **Nanda, S., Hacker, T. J. (2018).** RACC: Resource-aware container consolidation using a deep learning approach. Proceedings of the First Workshop on Machine Learning for Computing Systems, pp. 1–5. DOI: 10.1145/3217871.3217876.
18. **Li, F., Hu, B. (2019).** DeepJS: Job scheduling based on deep reinforcement learning in cloud data center. Proceedings of the 4th International Conference on Big Data and Computing. pp. 48–53. DOI: 10.1145/3335484.3335513.
19. **Verma, R., Singhal, A., Khadilkar, H., Basumatary, A., Nayak, S., Singh, H. V., Sinha, R. (2020).** A generalized reinforcement learning algorithm for online 3d bin-packing. arXiv:2007.00463. DOI: 10.48550/arXiv.2007.00463.
20. **Goyal, A., Deng, J. (2020).** Packit: A virtual environment for geometric planning. International Conference on Machine Learning Vol. 119, pp. 3700–3710.
21. **Silva-Gálvez, A., Orozco-Sanchez, J., Lara-Cárdenas, E., Ortiz-Bayliss, J. C., Amaya, I., Cruz-Duarte, J. M., Terashima-Marín, H. (2020).** Discovering action regions for solving the bin packing problem through hyper-heuristics. 2020 IEEE Symposium Series on Computational Intelligence, pp. 822–828. DOI: 10.1109/SSCI47803.2020.9308538.
22. **Zhao, H., She, Q., Zhu, C., Yang, Y., Xu, K. (2021).** Online 3D bin packing with constrained deep reinforcement learning. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 1, pp. 741–749. DOI: 10.1609/aaai.v35i1.16155.
23. **Yang, T., Luo, F., Fuentes, J., Ding, W., Gu, C. (2021).** A flexible reinforced bin packing framework with automatic slack selection. Mathematical Problems in Engineering, Vol. 2021, pp. 1-15. DOI: 10.1155/2021/6653586.

24. **Zhang, J., Zi, B., Ge, X. (2021).** Attend2pack: Bin packing through deep reinforcement learning with attention. arXiv:2107.04333. DOI: 10.48550/arXiv.2107.04333.
25. **Jiang, Y., Cao, Z., Zhang, J. (2021).** Solving 3D bin packing problem via multimodal deep reinforcement learning. Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, Vol. 3, pp. 1548–1550.
26. **Murdivien, S. A., Um, J. (2023).** BoxStacker: Deep reinforcement learning for 3D bin packing problem in virtual environment of logistics systems. Sensors, Vol. 23, No. 15, pp. 6928. DOI: 10.3390/s23156928.
27. **Fang, J., Rao, Y., Zhao, X., Du, B. (2023).** A hybrid reinforcement learning algorithm for 2D irregular packing problems. Mathematics, Vol. 11, No. 2, pp. 327. DOI: 10.3390/math 11020327.
28. **Guerriero, F., Saccomanno, F. P. (2023).** A machine learning approach for the bin packing problem. 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IEEE, Vol. 1, pp. 436–441. DOI: 10.1109/IDAACS58523.2023.10348703.
29. **Duan, L., Hu, H., Qian, Y., Gong, Y., Zhang, X., Xu, Y., Wei, J. (2018)**. A multi-task selected learning approach for solving 3D flexible bin packing problem. arXiv:1804.06896. DOI: 10.48550/arXiv.1804.06896.
30. **Mohiuddin, I., Almogren, A., Al Qurishi, M., Hassan, M. M., Al-Rassan, I., Fortino, G. (2019**). Secure distributed adaptive bin packing algorithm for cloud storage. Future Generation Computer Systems, Vol. 90, pp. 307–316. DOI: 10.1016/j.future.2018.08.013.
31. **Sim, K., Hart, E., Paechter, B. (2013).** Learning to solve bin packing problems with an immune inspired hyper-heuristic, pp. 856–863. DOI: 10.7551/978-0-262-31709-2-ch126.
32. **Sim, K., Hart, E. (2013).** Generating single and multiple cooperative heuristics for the one-dimensional bin packing problem using a single node genetic programming island model. Proceedings of the 15th annual conference on Genetic and evolutionary computation, pp. 1549–1556. DOI: 10.1145/ 2463372.2463555.
33. **Scholl, A., Klein, R., Jürgens, C. (1997).** Bison: A fast hybrid procedure for exactly solving the one-dimensional bin packing problem. Computers & Operations Research, Vol. 24, No. 7, pp. 627–645. DOI: 10.1016/ S0305-0548(96)00082-2.
34. **Falkenauer, E. (1996).** A hybrid grouping genetic algorithm for bin packing. Journal of Heuristics, Vol. 2, pp. 5–30. DOI: 10.1007/ BF00226291.
35. **Ha, C. T., Nguyen, T. T., Bui, L. T., Wang, R. (2017).** An online packing heuristic for the three-dimensional container loading problem in dynamic environments and the Physical Internet. Applications of Evolutionary Computation: 20th European Conference, EvoApplications 2017, Amsterdam, Springer, International Publishing, pp. 140–155. DOI: 10.1007/978-3-319-55792-2_10.

*Article received on 29/02/2024; accepted on 15/05/2024.*
**Corresponding author is Laura Cruz-Reyes.*

ISSN 2007-9737

# Experimental Analysis of a Cooperative Coevolutionary Algorithm with Parameter Tuning for Multi-objective Problem Optimization with Uncertainty

Lorena Rosas[1], Claudia Guadalupe Gómez Santillán[1, *], Nelson Rangel-Valdez[2], Maria Lucila Morales-Rodriguez[1], Héctor Fraire Huacuja[1], Manuel Vargas[1]

[1] National Technological Institute of Mexico,
Technological Institute of Madero,
Division of Postgraduate Studies and Research,
Mexico

[2] Technological Institute of Madero,
CONACyT- National Technological Institute of Mexico,
Mexico

{lorenarocio52 , jmvm100}@gmail.com,
{claudia.gs, nelson.rv, lucila.mr, hector.fh}@cdmadero.tecnm.mx

**Abstract.** Currently, organizations face significant challenges demanding effective and efficient solutions. The problem optimization and decision-making coupled with Decision Maker Preferences (DMPs), are crucial for achieving success and maintaining a competitive edge. In many cases, business problems involve the need to optimize multiple conflicting objectives, and DMPs may not be entirely precise. Coevolutionary algorithms have become increasingly popular as effective tools for solving problems involving multiple objectives. These techniques enable the simultaneous evolution of multiple solutions through the interaction and joint improve of different populations. Coevolutionary algorithms promote cooperative solution improvement, fostering diversity and facilitating the discovery of optimal solutions to complex problems. Parameter tuning is critical in coevolutionary algorithms as it determines how potential solutions are explored and enhances their ability to avoid local optima, directing the search toward global solutions. In this article, an analysis is conducted to identify the most viable configurations using parameter tuning in a cooperative coevolutionary algorithm to solve multi-objective problems with uncertainty. Experimental results demonstrate that no configuration dominates by absolute distance, but options are identified that can generate high-quality solutions.

**Keywords.** Parameter tuning, cooperative coevolution algorithm, multi-objective problem optimization.

## 1 Introduction

In business, scientific, and everyday life contexts, optimizing multi-objective problems has become a challenge spanning various disciplines, from engineering to decision-making in daily situations. In a world where efficiency is essential for success and competition is ever-increasing, the ability to optimize solutions that align with individual preferences, even when these preferences involve imprecise values.

Decision-making, as emphasized by Koziol [1], is one of the most critical processes in managing modern organizations due to its direct impact on the success or failure of the organization.

This aspect becomes even more relevant in environments characterized by uncertainty in information, where understanding the factors influencing decision-making is crucial for tackling challenges in problems. Currently, to address problems such as many-objective optimization problems (MaOP), coevolutionary strategies have been applied.

An example of this, is the work of Li et al. [2], who propose a cooperative coevolutionary algorithm with a dynamic learning strategy to solve

MaOP, called Dynamic Learning Cooperative Coevolutionary Algorithm of Two Populations (DL-TPCEA). In this work, they use only two populations to exchange information with evolution based on two criterias: one population uses the Pareto criterion (PC) and the other uses the non-Pareto criterion (NPC). Parameter tuning is performed only for the maximum number of iterations, the number of objectives, the diversity value, and the strategy of division in the coevolutionary approach by objectives.

On the other hand, coevolutionary strategies have also been employed to solve large-scale multi-objective optimization problems (LSMOP), such as in the work of Zhong, et al. [3], who proposed a cooperative coevolutionary algorithm using hybrid NSGA-II with Linkage Measurement Minimization (CC-HNSGA-LMM).

They apply an elitist genetic algorithm for the problem decomposition stage, then they move to an optimization stage and introduce the best-generated solution into the cooperative coevolutionary algorithm, employing a variable clustering method for LSMOP, aiming to improve solutions around the Pareto Front.

In this work, parameter adjustment is used in the clustering stage: the population size is 20, the maximum set of iterations is 20, and the gene size (number of clusters) is 6 and 7. They also perform parameter adjustments for subproblem optimization: dimensions of 500D and 1000D. Besides adjusting their genetic operators.

Upon analysis, similarities were found between the reviewed works and ours, such as the use of coevolutionary algorithms. However, unlike our work, they [2, 3] use the Pareto Front, whereas we address the Region of Interest (ROI). The ROI represents a set of solutions that are DMP, where we apply values with uncertainty to the DMP, allowing a range of values in these preferences.

Additionally, instead of finding a ROI, our approach generates a set of them. Furthermore, these works also employ parameter tuning, but with a different focus. In these works, parameter adjustment focuses on genetic operators such as crossover and mutation, whereas our approach focuses on configuring the number of species, the species division strategy (by variables and objectives), and the number of variables and objectives per species.

Until this date, classic evolutionary algorithms configure parameters of genetic operators such as crossover, mutation, and selection [3][4]. However, when addressing an evolutionary algorithm, they tend to propose a specific design in the values of their parameters, without considering small variations in them.

This research conducts a study on the impact on the quality of solutions to multi-objective problems when applying different coevolutionary designs. The results demonstrate that the parameters of coevolutionary design indeed have an impact on the quality of solutions and should be considered in the problem-solving process.

The contribution of this work is a methodology for designing coevolutionary algorithms, which is based on the configuration of three parameters: 1) the number of species, 2) the division strategy (by objectives or variables), and 3) the number of variables and objectives per species.

This article consists of the following sections: Section 2 provides general information, including a description of multi-objective optimization under uncertainty, a definition of the coevolutionary algorithm, and a description of the cooperative coevolutionary approach by Potter and De Jong [5], as well as details on parameter tuning. Section 3 addresses the solution methodology, while Section 4 describes the experimental design. The results of the experimentation are discussed in Section 5, and finally, a general conclusion is presented in Section 6.

## 2 Background

### 2.1 Multi-objective Optimization with Uncertainty

Within the realm of multi-objective optimization, the fundamental task is to determine values for a set of decision variables that satisfy constraints while optimizing a vectorial function [6]. This vectorial function is composed of elements representing individual objective functions, which offer a mathematical representation of various performance criteria that often conflict with each other. Multi-objective optimization aims to find a solution that provides acceptable values for all objective functions for the decision-maker.

This approach seeks to reconcile objectives and constraints, considering potential contradictions among the stated objectives [7, 8]. Precise mathematical formulation is crucial to address this challenge and reflect the complexity of optimization criteria and involved constraints (Eq. 1).

The decision-maker (DM) can express each objective as an interval, considering imprecision, that is, $f_j(x) = \left[\underline{f_j}(x), \overline{f_j}(x)\right]$Each element of the set X is treated as a vector of intervals $\vec{f}(x)$:

$$max\ F(x) = (f_1, f_2, \ldots, f_m) \quad \text{Subject to: } x \in \Omega. \tag{1}$$

Uncertainty arises from incomplete knowledge. Representing objectives as intervals reflect the imprecision associated with the goals, enabling greater flexibility in decision-making in the face of uncertainty. This approach incorporates the inherent variability in the decision-making process and enhances the model's ability to adapt to situations where available information may be imprecise.

### 2.2 Coevolutionary Algorithm

Coevolution represents a process through which two or more species interact mutually and undergo evolutionary changes in response to adaptations observed in each other. This concept originates from the principles of Charles Darwin in 1859, where the notion of evolution was introduced [9].

Within coevolutionary algorithms, two predominant approaches are distinguished: cooperation and competition. In the cooperative approach, populations work together to enhance overall performance [10], whereas in the competitive approach, populations compete to gain advantages over other populations. This research work uses a cooperative approach as the central focus of the study.

### 2.3 Cooperative Coevolutionary Approach

In 1994, Potter and De Jong introduced the technique of cooperative coevolution (Fig. 1) to address challenges in function optimization [5]. This strategy involves decomposing the objective function into several sub-functions, allowing the evolution of independent solutions for each of them. Subsequently, these solutions are combined into a complete solution.

Cooperative coevolution directs the evolutionary process towards more manageable subproblems, avoiding to get trapped in the local optima of the global function.

Furthermore, it promotes learning among solutions, resulting in continuous performance improvement over time. Currently, numerous research studies have successfully implemented coevolution in algorithms to address a variety of challenges. These studies have highlighted the potential of coevolutionary algorithms in solving complex problems in diverse fields such as engineering, optimization, and decision-making, among others [2, 11, 12].

Despite their advantages, it is crucial to remember that achieving successful results in coevolutionary algorithms largely depends on the configuration of their parameters. Adjustment parameters, such as population size, mutation and crossing rate, and the number of generations provide a critical factor to achieve efficient performance and obtain high quality solutions. Bad parameter options could lead to premature convergence or insufficient diversity in the quality of solutions.

## 3 Methodology

The Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) along with its various variants from VAR1 to VAR6 proposed by Fernandez, et al. [13] has been incorporated into a cooperative coevolutionary algorithm. The design of Fernandez's algorithm [14] is aimed at addressing the imprecision present in the data used by the DM by incorporating interval-based models and dominance relationships. The variants introduce modifications to the Tchebychev scalar function of the MOEA/D to integrate dominance relationships into the evolutionary search process.

Mejía [15] identifies two perspectives in parameter tuning: the "online" approach, where values are dynamically updated during algorithm execution, and the "offline" approach, where it is done before algorithm execution.

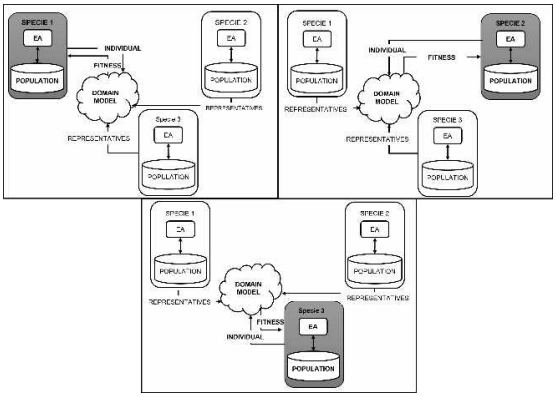



**Fig. 1.** Coevolutionary model of three species shown from the perspective of each in turn [5]

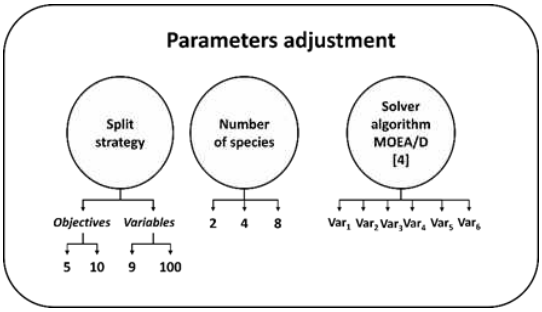



**Fig. 2.** Cooperative coevolutive parameter adjustment

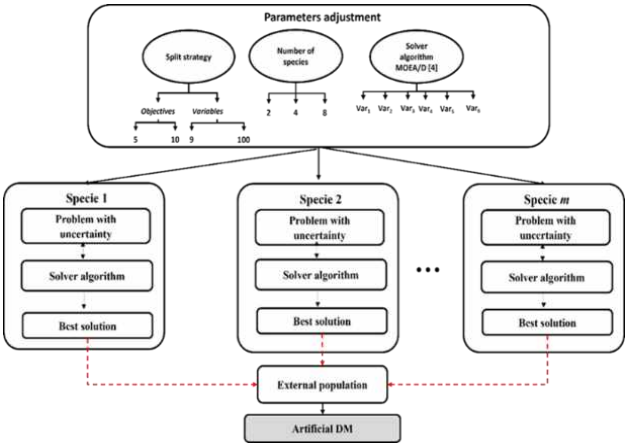



**Fig. 3.** Cooperative coevolutive algorithm with uncertainty

Parameter tuning is crucial as it significantly contributes to the efficiency and effectiveness of the optimization process, allowing for high-quality results in solution search.

In Fig. 2, different parameters adjusted in the coevolutionary algorithm can be observed, such as the split strategy which can be based on objectives or variables, the number of species the algorithm will work with, and the solving algorithm for the species.

Fig. 3 illustrates the cooperative coevolutionary algorithm, where parameter tuning is done offline, as it takes place before the coevolutionary process begins.

**Table 1.** Experimental design of coevolutionary algorithms

| Algorithm | Variant | Split strategy |
|---|---|---|
| Coevolutionary 1 | | 2 objectives |
| Coevolutionary 2 | | 4 objectives |
| Coevolutionary 3 | MOEA/D | 2 variables |
| Coevolutionary 4 | | 4 variables |
| Coevolutionary 5 | | 8 variables |
| Coevolutionary 6 | | 2 objectives |
| Coevolutionary 7 | | 4 objectives |
| Coevolutionary 8 | VAR1 | 2 variables |
| Coevolutionary 9 | | 4 variables |
| Coevolutionary 10 | | 8 variables |
| Coevolutionary 11 | | 2 objectives |
| Coevolutionary 12 | | 4 objectives |
| Coevolutionary 13 | VAR2 | 2 variables |
| Coevolutionary 14 | | 4 variables |
| Coevolutionary 15 | | 8 variables |
| Coevolutionary 16 | | 2 objectives |
| Coevolutionary 17 | | 4 objectives |
| Coevolutionary 18 | VAR3 | 2 variables |
| Coevolutionary 19 | | 4 variables |
| Coevolutionary 20 | | 8 variables |
| Coevolutionary 21 | | 2 objectives |
| Coevolutionary 22 | | 4 objectives |
| Coevolutionary 23 | VAR4 | 2 variables |
| Coevolutionary 24 | | 4 variables |
| Coevolutionary 25 | | 8 variables |
| Coevolutionary 26 | | 2 objectives |
| Coevolutionary 27 | | 4 objectives |
| Coevolutionary 28 | VAR5 | 2 variables |
| Coevolutionary 29 | | 4 variables |
| Coevolutionary 30 | | 8 variables |
| Coevolutionary 31 | | 2 objectives |
| Coevolutionary 32 | | 4 objectives |
| Coevolutionary 33 | VAR6 | 2 variables |
| Coevolutionary 34 | | 4 variables |
| Coevolutionary 35 | | 8 variables |

Having the parameter values, the problem is divided among the number of species, with each species working on the respective objectives or variables to evolve. It then moves to the solving algorithm, which receives the species to work with and generates a population subjected to mutation and crossover operators.

Consequently, each species selects a representative, which is the best solution of the species. In the end, each species cooperates with each other, forming complete solutions stored in the external populations (EP).

## 4 Experimental Design

DTLZ1-7 and WFG1-9 are benchmark problems that have been used to assess the performance of MOEA/D with preferences. The number of objectives used in this study was m = {5, 10} and the number of variables used was k= {9,100}.

### 4.1 Parameters Tuning

When addressing real-world problems, it is common to use algorithms that require specific configurations to ensure competitive performance. The quality of an algorithm is closely linked to the values of its parameters, posing a crucial challenge in various fields. In his article, Ocaño et al. [16] point out the No Free Lunch (NFL) theorem, which states that there is no set of optimal parameters that can solve all optimization problems.

Therefore, parameter tuning is essential for algorithm performance, requiring careful selection of values to guide its behavior and performance.

In the context of coevolutionary algorithms, this process involves selecting appropriate values for specific elements such as configuring the number of species, the species division strategy (by variables and objectives), and the number of variables and objectives per species.

In this study, we focus on offline parameter tuning, where the configuration is established before the algorithm execution, recognizing the importance of this phase in obtaining effective results in real-world problems.

**Table 2.** Results of best parameter configurations for problems with five and ten objectives

| No. objectives | Variant | Split strategy | No. species |
|---|---|---|---|
| 5 | $VAR_2$ | Variables | 4 |
| | $VAR_6$ | Variables | 4 |
| 10 | $VAR_6$ | Objectives | 4 |
| | $VAR_4$ | Variables | 4 |

### 4.2 Quality Measures

Quality measures, also known as metrics or indicators, play a fundamental role in evaluating the efficiency and performance of algorithms, especially in the context of optimization. In the study by Castellanos [17], it is noted that while traditional metrics such as hypervolume, spacing, and dispersion exist, these may be insufficient when evaluating preference-based algorithms.

In that work, the use of distance-based indicators to measure quality in terms of similarity between the optimal set $X^*$and the Approximation to the Region of Interest (A-RoI) is proposed. The minimum Euclidean distance and the minimum Tchebychev distance are used to calculate the proximity of the A-RoI to the nearest point in $X^*$.

Additionally, the average of minimum Euclidean distance and the average of minimum Tchebychev distance are introduced, representing the average distance from points in the A-RoI to those in $X^*$. Furthermore, the median Euclidean and median Tchebychev are considered, reflecting the median distances from the A-RoI to the nearest point in $X^*$.

Each of these metrics underwent the Borda Count method to determine the best configuration according to the DMPs. This voting method assigns scores to options based on their ranking by each voter. The steps to perform the Borda Count are as follows:

1 **Assigning Scores**: Each candidate receives a score based on their position in each preference list. The most preferred candidate in a list receives the highest score, and scores decrease as you move down the list.
2 **Assigning Scores to Preferences:** A specific score is assigned to each position in the

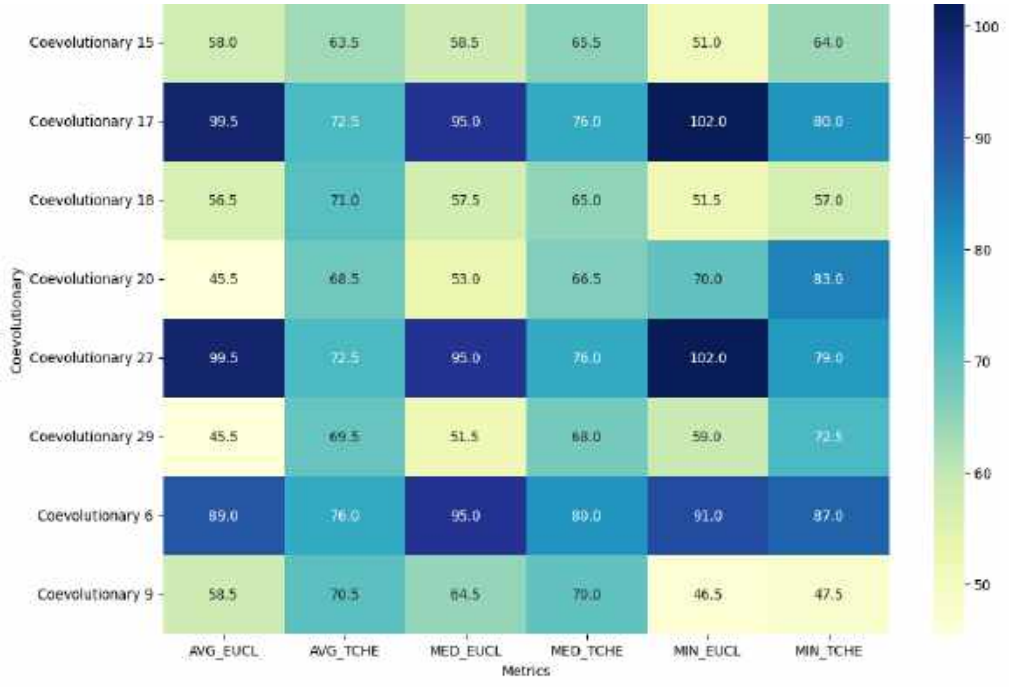



**Fig. 4.** Counting summary of algorithms with good performance on five objective problems

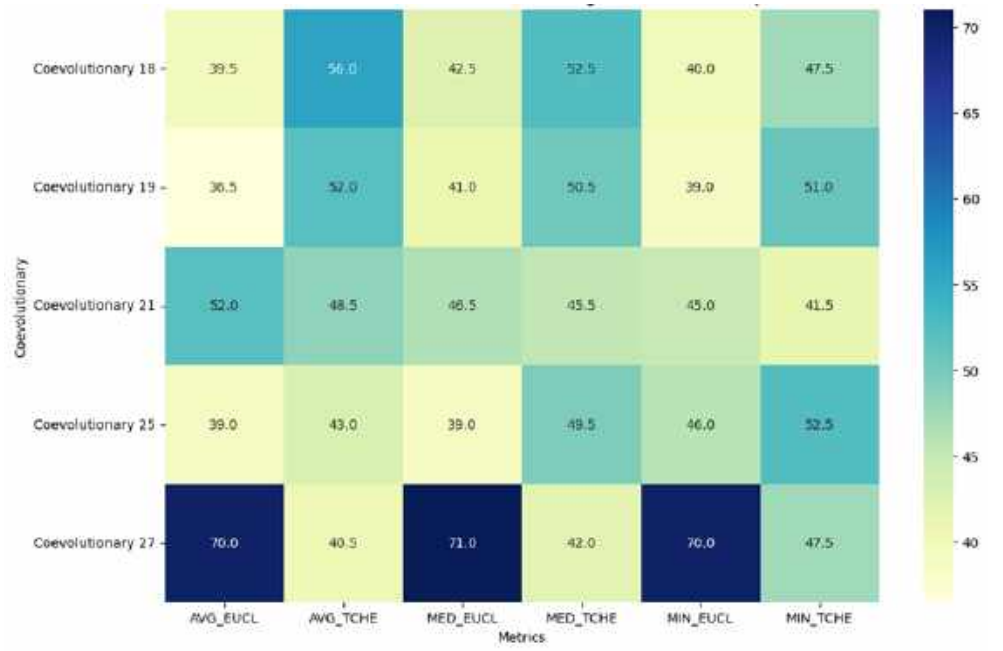



**Fig. 5.** counting summary of algorithms with good performance on ten objective problems

preference list. For example, if there are n candidates, the most preferred candidate in a list may receive *n* points, the next one *n*−1 points, and so forth.

**Table 3.** Ranking of problem with five objectives, division of objectives and two species

| | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| **PROBLEMS WITH FIVE OBJECTIVES. DIVISION BY OBJECTIVES AND TWO SPECIES** | **DTLZ1** | MOEA/D | 4 | 5 | 5.5 | 5 | 5 | 5 |
| | | $VAR_1$ | 1 | 1.5 | 1 | 1.5 | 1 | 1 |
| | | $VAR_2$ | 4 | 5 | 5.5 | 5 | 5 | 5 |
| | | $VAR_3$ | 6.5 | 5 | 5.5 | 5 | 5 | 5 |
| | | $VAR_4$ | 2 | 1.5 | 2 | 1.5 | 2 | 2 |
| | | $VAR_5$ | 6.5 | 5 | 5.5 | 5 | 5 | 5 |
| | | $VAR_6$ | 4 | 5 | 3 | 5 | 5 | 5 |
| | **DTLZ2** | MOEA/D | 6 | 6 | 4 | 6 | 6 | 4.5 |
| | | $VAR_1$ | 3 | 2.5 | 4 | 3 | 2.5 | 4.5 |
| | | $VAR_2$ | 6 | 6 | 4 | 6 | 6 | 1 |
| | | $VAR_3$ | 3 | 2.5 | 4 | 3 | 2.5 | 4.5 |
| | | $VAR_4$ | 3 | 2.5 | 4 | 3 | 2.5 | 4.5 |
| | | $VAR_5$ | 1 | 2.5 | 4 | 1 | 2.5 | 4.5 |
| | | $VAR_6$ | 6 | 6 | 4 | 6 | 6 | 4.5 |
| | **DTLZ3** | MOEA/D | 3 | 6 | 3 | 6 | 3 | 2.5 |
| | | $VAR_1$ | 3 | 2.5 | 3 | 1 | 3 | 2.5 |
| | | $VAR_2$ | 3 | 2.5 | 3 | 3 | 3 | 2.5 |
| | | $VAR_3$ | 6.5 | 6 | 6.5 | 6 | 6.5 | 6.5 |
| | | $VAR_4$ | 3 | 2.5 | 3 | 3 | 3 | 2.5 |
| | | $VAR_5$ | 6.5 | 6 | 6.5 | 6 | 6.5 | 6.5 |
| | | $VAR_6$ | 3 | 2.5 | 3 | 3 | 3 | 5 |
| | **DTLZ4** | MOEA/D | 3 | 6 | 4.5 | 6 | 3 | 4.5 |
| | | $VAR_1$ | 6.5 | 2.5 | 1 | 2.5 | 6.5 | 1 |
| | | $VAR_2$ | 3 | 6 | 4.5 | 6 | 3 | 4.5 |
| | | $VAR_3$ | 3 | 2.5 | 4.5 | 2.5 | 3 | 4.5 |
| | | $VAR_4$ | 6.5 | 2.5 | 4.5 | 2.5 | 6.5 | 4.5 |
| | | $VAR_5$ | 3 | 2.5 | 4.5 | 2.5 | 3 | 4.5 |
| | | $VAR_6$ | 3 | 6 | 4.5 | 6 | 3 | 4.5 |
| | **DTLZ5** | MOEA/D | 7 | 2 | 6 | 2 | 7 | 7 |
| | | $VAR_1$ | 3.5 | 5.5 | 3.5 | 5.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 3.5 | 2 | 6 | 2 | 3.5 | 3.5 |
| | | $VAR_3$ | 3.5 | 5.5 | 3.5 | 5.5 | 3.5 | 3.5 |
| | | $VAR_4$ | 3.5 | 5.5 | 1.5 | 5.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 3.5 | 5.5 | 1.5 | 5.5 | 3.5 | 3.5 |
| | | $VAR_6$ | 3.5 | 2 | 6 | 2 | 3.5 | 3.5 |
| | **DTLZ6** | MOEA/D | 5 | 5 | 5 | 2 | 4 | 4 |
| | | $VAR_1$ | 1.5 | 5 | 1.5 | 5.5 | 4 | 4 |
| | | $VAR_2$ | 5 | 1.5 | 5 | 2 | 4 | 4 |
| | | $VAR_3$ | 5 | 5 | 5 | 5.5 | 4 | 4 |
| | | $VAR_4$ | 1.5 | 5 | 1.5 | 5.5 | 4 | 4 |
| | | $VAR_5$ | 5 | 5 | 5 | 5.5 | 4 | 4 |
| | | $VAR_6$ | 5 | 1.5 | 5 | 2 | 4 | 4 |
| | **DTLZ7** | MOEA/D | 4 | 1 | 1 | 1 | 4 | 4 |
| | | $VAR_1$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4 |
| | | $VAR_2$ | 4 | 1 | 4.5 | 3 | 4 | 4 |
| | | $VAR_3$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4 |
| | | $VAR_4$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4 |
| | | $VAR_5$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4 |
| | | $VAR_6$ | 4 | 1 | 4.5 | 2 | 4 | 4 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| WFG1 | MOEA/D | 3 | 5.5 | 6 | 6 | 3 | 6 |
| | $VAR_1$ | 3 | 1 | 2.5 | 2.5 | 3 | 2.5 |
| | $VAR_2$ | 3 | 1 | 6 | 6 | 3 | 6 |
| | $VAR_3$ | 6.5 | 5.5 | 2.5 | 2.5 | 6.5 | 2.5 |
| | $VAR_4$ | 3 | 1 | 2.5 | 2.5 | 3 | 2.5 |
| | $VAR_5$ | 6.5 | 5.5 | 2.5 | 2.5 | 6.5 | 2.5 |
| | $VAR_6$ | 3 | 5.5 | 6 | 6 | 3 | 6 |
| WFG2 | MOEA/D | 5.5 | 4 | 1 | 4 | 6 | 2 |
| | $VAR_1$ | 2.5 | 4 | 6 | 4 | 4 | 5.5 |
| | $VAR_2$ | 5.5 | 4 | 2.5 | 4 | 6 | 2 |
| | $VAR_3$ | 5.5 | 4 | 6 | 4 | 2 | 5.5 |
| | $VAR_4$ | 1 | 4 | 4 | 4 | 2 | 5.5 |
| | $VAR_5$ | 2.5 | 4 | 6 | 4 | 2 | 5.5 |
| | $VAR_6$ | 5.5 | 4 | 2.5 | 4 | 6 | 2 |
| WFG3 | MOEA/D | 6 | 5 | 5 | 4.5 | 6 | 5 |
| | $VAR_1$ | 2.5 | 1.5 | 5 | 1 | 2 | 5 |
| | $VAR_2$ | 6 | 5 | 5 | 4.5 | 6 | 5 |
| | $VAR_3$ | 2.5 | 5 | 1.5 | 4.5 | 2 | 2 |
| | $VAR_4$ | 2.5 | 1.5 | 5 | 4.5 | 4 | 5 |
| | $VAR_5$ | 2.5 | 5 | 1.5 | 4.5 | 2 | 1 |
| | $VAR_6$ | 6 | 5 | 5 | 4.5 | 6 | 5 |
| WFG4 | MOEA/D | 6 | 6 | 2 | 6 | 6 | 2 |
| | $VAR_1$ | 2.5 | 2.5 | 5.5 | 3.5 | 3 | 5.5 |
| | $VAR_2$ | 6 | 6 | 2 | 6 | 6 | 2 |
| | $VAR_3$ | 2.5 | 2.5 | 5.5 | 2 | 3 | 5.5 |
| | $VAR_4$ | 2.5 | 2.5 | 5.5 | 3.5 | 3 | 5.5 |
| | $VAR_5$ | 2.5 | 2.5 | 5.5 | 1 | 1 | 5.5 |
| | $VAR_6$ | 6 | 6 | 2 | 6 | 6 | 2 |
| WFG5 | MOEA/D | 3 | 6 | 6 | 6 | 3 | 6 |
| | $VAR_1$ | 3 | 2.5 | 2.5 | 2.5 | 3 | 4 |
| | $VAR_2$ | 6.5 | 6 | 6 | 6 | 6.5 | 6 |
| | $VAR_3$ | 3 | 2.5 | 2.5 | 2.5 | 3 | 2 |
| | $VAR_4$ | 3 | 2.5 | 2.5 | 2.5 | 3 | 2 |
| | $VAR_5$ | 3 | 2.5 | 2.5 | 2.5 | 3 | 2 |
| | $VAR_6$ | 6.5 | 6 | 6 | 6 | 6.5 | 6 |
| WFG6 | MOEA/D | 5.5 | 5.5 | 4.5 | 5 | 5 | 4.5 |
| | $VAR_1$ | 5.5 | 5.5 | 4.5 | 5 | 5 | 4.5 |
| | $VAR_2$ | 5.5 | 5.5 | 4.5 | 5 | 5 | 4.5 |
| | $VAR_3$ | 2 | 2 | 4.5 | 2 | 1.5 | 4.5 |
| | $VAR_4$ | 3 | 2 | 4.5 | 5 | 5 | 4.5 |
| | $VAR_5$ | 1 | 2 | 1 | 1 | 1.5 | 1 |
| | $VAR_6$ | 5.5 | 5.5 | 4.5 | 5 | 5 | 4.5 |
| WFG7 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_1$ | 3 | 4 | 2.5 | 2.5 | 3 | 2.5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_3$ | 2 | 1 | 2.5 | 2.5 | 2 | 2.5 |
| | $VAR_4$ | 4 | 1 | 2.5 | 2.5 | 4 | 2.5 |
| | $VAR_5$ | 1 | 1 | 2.5 | 2.5 | 1 | 2.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| WFG8 | MOEA/D | 6 | 6 | 4 | 6 | 6 | 4 |
| | $VAR_1$ | 2.5 | 2.5 | 4 | 2.5 | 3.5 | 4 |
| | $VAR_2$ | 6 | 6 | 4 | 6 | 6 | 4 |
| | $VAR_3$ | 2.5 | 2.5 | 4 | 2.5 | 3.5 | 4 |
| | $VAR_4$ | 2.5 | 2.5 | 4 | 2.5 | 1.5 | 4 |
| | $VAR_5$ | 2.5 | 2.5 | 4 | 2.5 | 1.5 | 4 |
| | $VAR_6$ | 6 | 6 | 4 | 6 | 6 | 4 |
| WFG9 | MOEA/D | 4.5 | 4.5 | 4 | 4 | 5 | 4 |
| | $VAR_1$ | 1 | 4.5 | 4 | 4 | 1 | 4 |
| | $VAR_2$ | 4.5 | 4.5 | 4 | 4 | 5 | 4 |
| | $VAR_3$ | 4.5 | 4.5 | 4 | 4 | 5 | 4 |
| | $VAR_4$ | 4.5 | 4.5 | 4 | 4 | 2 | 4 |
| | $VAR_5$ | 4.5 | 1 | 4 | 4 | 5 | 4 |
| | $VAR_6$ | 4.5 | 4.5 | 4 | 4 | 5 | 4 |

3 **Summing Scores:** The scores for each candidate from all lists are added. The candidate with the highest total score is considered the winner if the problem is a maximization problem, but if it is a minimization problem, the candidate with the lowest total score is considered the winner.

ISSN 2007-9737



**Table 4**. Borda count of coevolutionary with objective division and two species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 77.5 | 79.5 | 67.5 | 75.5 | 78 | 71 |
| **VAR1** | 48 | 53 | 55 | 52 | 52 | 58 |
| **VAR2** | 77.5 | 68 | 72.5 | 74.5 | 78 | 64 |
| **VAR3** | 62.5 | 61.5 | 66.5 | 59.5 | 57 | 64.5 |
| **VAR4** | 49.5 | 46.5 | 55.5 | 57.5 | 53 | 60.5 |
| **VAR5** | 55.5 | 58 | 61 | 55.5 | 52 | 60 |
| **VAR6** | 77.5 | 72.5 | 70 | 73.5 | 78 | 70 |

**Table 5.** Ranking of problem with five objectives, division of objectives and four species.

| PROBLEMS WITH FIVE OBJECTIVES, DIVISION BY OBJECTIVES AND FOUR SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 2 | 5 | 2 | 6 | 2 | 2 |
| | | $VAR_1$ | 5.5 | 5 | 5.5 | 2.5 | 5.5 | 5.5 |
| | | $VAR_2$ | 2 | 5 | 2 | 6 | 2 | 2 |
| | | $VAR_3$ | 5.5 | 2 | 5.5 | 2.5 | 5.5 | 5.5 |
| | | $VAR_4$ | 5.5 | 5 | 5.5 | 2.5 | 5.5 | 5.5 |
| | | $VAR_5$ | 5.5 | 1 | 5.5 | 2.5 | 5.5 | 5.5 |
| | | $VAR_6$ | 2 | 5 | 2 | 6 | 2 | 2 |
| | DTLZ2 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ3 | MOEA/D | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_1$ | 6 | 5.5 | 5.5 | 5.5 | 6 | 5.5 |
| | | $VAR_2$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_3$ | 6 | 5.5 | 5.5 | 5.5 | 6 | 5.5 |
| | | $VAR_4$ | 4 | 5.5 | 5.5 | 5.5 | 4 | 5.5 |
| | | $VAR_5$ | 6 | 5.5 | 5.5 | 5.5 | 6 | 5.5 |
| | | $VAR_6$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | DTLZ4 | MOEA/D | 5 | 6 | 6 | 6 | 5 | 5 |
| | | $VAR_1$ | 5 | 2.5 | 4 | 2.5 | 5 | 5 |
| | | $VAR_2$ | 5 | 6 | 6 | 6 | 5 | 5 |
| | | $VAR_3$ | 1.5 | 2.5 | 1 | 2.5 | 1.5 | 1 |
| | | $VAR_4$ | 5 | 2.5 | 3 | 2.5 | 5 | 5 |
| | | $VAR_5$ | 1.5 | 2.5 | 2 | 2.5 | 1.5 | 2 |
| | | $VAR_6$ | 5 | 6 | 6 | 6 | 5 | 5 |
| | DTLZ5 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ6 | MOEA/D | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_1$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_2$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_3$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_4$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_5$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_6$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | DTLZ7 | MOEA/D | 2 | 1 | 2 | 4 | 4 | 4 |
| | | $VAR_1$ | 5 | 5.5 | 5.5 | 4 | 4 | 4 |
| | | $VAR_2$ | 1 | 1 | 2 | 4 | 4 | 4 |
| | | $VAR_3$ | 5 | 5.5 | 5.5 | 4 | 4 | 4 |
| | | $VAR_4$ | 5 | 5.5 | 5.5 | 4 | 4 | 4 |
| | | $VAR_5$ | 5 | 5.5 | 5.5 | 4 | 4 | 4 |
| | | $VAR_6$ | 5 | 1 | 2 | 4 | 4 | 4 |
| | WFG1 | MOEA/D | 2 | 2 | 3 | 5 | 2 | 3 |
| | | $VAR_1$ | 5 | 5.5 | 3 | 1.5 | 5.5 | 3 |
| | | $VAR_2$ | 5 | 2 | 3 | 5 | 2 | 3 |
| | | $VAR_3$ | 5 | 5.5 | 6.5 | 5 | 5.5 | 6.5 |
| | | $VAR_4$ | 5 | 5.5 | 3 | 1.5 | 5.5 | 3 |
| | | VAR5 | 5 | 5.5 | 6.5 | 5 | 5.5 | 6.5 |
| | | VAR6 | 1 | 2 | 3 | 5 | 2 | 3 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG2** | MOEA/D | 4 | 3 | 2 | 2.5 | 3 | 1 |
| | $VAR_1$ | 4 | 3 | 5.5 | 5.5 | 3 | 5.5 |
| | $VAR_2$ | 2 | 3 | 2 | 1 | 3 | 1 |
| | $VAR_3$ | 6.5 | 6.5 | 5.5 | 5.5 | 6.5 | 5.5 |
| | $VAR_4$ | 4 | 3 | 5.5 | 5.5 | 3 | 5.5 |
| | $VAR_5$ | 6.5 | 6.5 | 5.5 | 5.5 | 6.5 | 5.5 |
| | $VAR_6$ | 1 | 3 | 2 | 2.5 | 3 | 1 |
| **WFG3** | MOEA/D | 6 | 6 | 6 | 6 | 5 | 5 |
| | $VAR_1$ | 3.5 | 2.5 | 3 | 2.5 | 5 | 5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 5 | 5 |
| | $VAR_3$ | 1 | 2.5 | 1 | 2.5 | 1.5 | 1.5 |
| | $VAR_4$ | 3.5 | 2.5 | 3 | 2.5 | 5 | 5 |
| | $VAR_5$ | 2 | 2.5 | 3 | 2.5 | 1.5 | 1.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 5 | 5 |
| **WFG4** | MOEA/D | 6 | 6 | 4 | 4 | 6 | 2 |
| | $VAR_1$ | 2.5 | 2.5 | 4 | 4 | 2.5 | 5 |
| | $VAR_2$ | 6 | 6 | 4 | 4 | 6 | 1 |
| | $VAR_3$ | 2.5 | 2.5 | 4 | 4 | 2.5 | 5 |
| | $VAR_4$ | 2.5 | 2.5 | 4 | 4 | 2.5 | 5 |
| | $VAR_5$ | 2.5 | 2.5 | 4 | 4 | 2.5 | 5 |
| | $VAR_6$ | 6 | 6 | 4 | 4 | 6 | 5 |
| **WFG5** | MOEA/D | 2 | 6 | 6 | 6 | 4 | 6 |
| | $VAR_1$ | 5.5 | 2.5 | 2.5 | 2.5 | 4 | 2.5 |
| | $VAR_2$ | 2 | 6 | 6 | 6 | 4 | 6 |
| | $VAR_3$ | 5.5 | 2.5 | 2.5 | 2.5 | 4 | 2.5 |
| | $VAR_4$ | 5.5 | 2.5 | 2.5 | 2.5 | 4 | 2.5 |
| | $VAR_5$ | 5.5 | 2.5 | 2.5 | 2.5 | 4 | 2.5 |
| | $VAR_6$ | 2 | 6 | 6 | 6 | 4 | 6 |
| **WFG6** | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 4 |
| | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2 |
| | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| **WFG7** | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| **WFG8** | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 3.5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 3.5 |
| | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 1.5 |
| | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 1.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| **WFG9** | MOEA/D | 6 | 6 | 6 | 6 | 4.5 | 5.5 |
| | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 1 | 2.5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 4.5 | 5.5 |
| | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 4.5 | 5.5 |
| | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 4.5 | 1 |
| | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 4.5 | 2.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 4.5 | 5.5 |

### 4.3 Experimental Conditions

Table 1 presents the experimental conditions used to validate the performance of coevolutionary algorithms with their different configurations and variants in solving standard problems.

The main objective was to identify the most effective or viable strategy that approximated the region of interest of the DM. The genetic operators employed were: random parent selection, SBX crossover, and polynomial-based mutation.

**Table 6.** Borda count of CO-MOEA/D with objective division and four species

| MIN_EUCL | | | | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|
| 75 | | | | 71 | 79.5 | 69.5 | 67.5 |
| 55 | | | | 59 | 51 | 59.5 | 62 |
| 75 | | | | 71 | 78 | 69.5 | 66.5 |
| 55.5 | 57.5 | 54.5 | 59.5 | 63 | | | |
| 55 | 58 | 51 | 61 | 58.5 | | | |
| 54.5 | 60.5 | 54.5 | 59.5 | 57 | | | |
| 75 | 71 | 79.5 | 69.5 | 70.5 | | | |

**Table 7**. Ranking of problem with five objectives, division of objectives and two species

| | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| PROBLEMS WITH FIVE OBJECTIVES. DIVISION BY OBJECTIVES AND TWO SPECIES | DTLZ1 | MOEA/D | 6 | 6 | 5.5 | 6 | 6 | 6 |
| | | $VAR_1$ | 3.5 | 3.5 | 3 | 3.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 6 | 1.5 | 5.5 | 1.5 | 6 | 6 |
| | | $VAR_3$ | 1 | 6 | 1 | 6 | 1.5 | 1.5 |
| | | $VAR_4$ | 3.5 | 3.5 | 5.5 | 3.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 2 | 6 | 2 | 6 | 1.5 | 1.5 |
| | | $VAR_6$ | 6 | 1.5 | 5.5 | 1.5 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 7 | 6 | 4.5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 3.5 | 2.5 | 4.5 | 4 | 2.5 |
| | | $VAR_2$ | 6 | 3.5 | 6 | 1 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 3.5 | 2.5 | 4.5 | 2 | 2.5 |
| | | $VAR_4$ | 2.5 | 3.5 | 2.5 | 4.5 | 2 | 2.5 |
| | | $VAR_5$ | 2.5 | 3.5 | 2.5 | 4.5 | 2 | 2.5 |
| | | $VAR_6$ | 6 | 3.5 | 6 | 4.5 | 6 | 6 |
| | DTLZ3 | MOEA/D | 5 | 4 | 5 | 4.5 | 5.5 | 5 |
| | | $VAR_1$ | 5 | 4 | 5 | 4.5 | 5.5 | 5 |
| | | $VAR_2$ | 5 | 4 | 5 | 4.5 | 5.5 | 5 |
| | | $VAR_3$ | 1.5 | 4 | 1.5 | 4.5 | 1 | 2 |
| | | $VAR_4$ | 5 | 4 | 5 | 4.5 | 1 | 5 |
| | | $VAR_5$ | 1.5 | 4 | 1.5 | 4.5 | 1 | 1 |
| | | $VAR_6$ | 5 | 4 | 5 | 1 | 5.5 | 5 |
| | DTLZ4 | MOEA/D | 5 | 4 | 5 | 3 | 5 | 5.5 |
| | | $VAR_1$ | 5 | 4 | 5 | 3 | 5 | 1 |
| | | $VAR_2$ | 5 | 4 | 5 | 3 | 5 | 5.5 |
| | | $VAR_3$ | 1.5 | 4 | 1.5 | 6.5 | 1.5 | 1 |
| | | $VAR_4$ | 5 | 4 | 5 | 3 | 5 | 5.5 |
| | | $VAR_5$ | 1.5 | 4 | 1.5 | 6.5 | 1.5 | 1 |
| | | $VAR_6$ | 5 | 4 | 5 | 3 | 5 | 5.5 |
| | DTLZ5 | MOEA/D | 7 | 1 | 7 | 1 | 5.5 | 7 |
| | | $VAR_1$ | 3.5 | 3 | 3.5 | 4.5 | 2 | 4 |
| | | $VAR_2$ | 3.5 | 5.5 | 3.5 | 4.5 | 2 | 4 |
| | | $VAR_3$ | 3.5 | 5.5 | 3.5 | 4.5 | 5.5 | 4 |
| | | $VAR_4$ | 3.5 | 5.5 | 3.5 | 4.5 | 5.5 | 4 |
| | | $VAR_5$ | 3.5 | 5.5 | 3.5 | 4.5 | 5.5 | 4 |
| | | $VAR_6$ | 3.5 | 2 | 3.5 | 4.5 | 2 | 1 |
| | DTLZ6 | MOEA/D | 7 | 1 | 7 | 1 | 6.5 | 6.5 |
| | | $VAR_1$ | 4.5 | 2 | 4 | 2 | 4.5 | 4.5 |
| | | $VAR_2$ | 1 | 6 | 1 | 6 | 1 | 1 |
| | | $VAR_3$ | 4.5 | 6 | 4 | 6 | 4.5 | 4.5 |
| | | $VAR_4$ | 4.5 | 3.5 | 4 | 3.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 4.5 | 6 | 4 | 6 | 6.5 | 6.5 |
| | | $VAR_6$ | 2 | 3.5 | 4 | 3.5 | 2.5 | 2.5 |
| | DTLZ7 | MOEA/D | 6 | 5 | 6 | 1 | 6 | 5.5 |
| | | $VAR_1$ | 3.5 | 5 | 3 | 5.5 | 1 | 2 |
| | | $VAR_2$ | 6 | 2 | 6 | 1 | 6 | 5.5 |
| | | $VAR_3$ | 1.5 | 5 | 3 | 5.5 | 1 | 2 |
| | | $VAR_4$ | 1.5 | 5 | 1 | 5.5 | 1 | 2 |
| | | $VAR_5$ | 3.5 | 5 | 3 | 5.5 | 4 | 5.5 |
| | | $VAR_6$ | 6 | 1 | 6 | 1 | 6 | 5.5 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | MOEA/D | 4 | 4 | 7 | 6.5 | 4 | 7 |
| | $VAR_1$ | 4 | 4 | 3.5 | 3 | 4 | 3.5 |
| | $VAR_2$ | 4 | 4 | 3.5 | 6.5 | 4 | 3.5 |
| | $VAR_3$ | 4 | 4 | 3.5 | 3 | 4 | 3.5 |
| | $VAR_4$ | 4 | 4 | 3.5 | 3 | 4 | 3.5 |
| | $VAR_5$ | 4 | 4 | 3.5 | 3 | 4 | 3.5 |
| | $VAR_6$ | 4 | 4 | 3.5 | 3 | 4 | 3.5 |
| **WFG2** | MOEA/D | 6.5 | 4 | 1 | 1 | 7 | 4 |
| | $VAR_1$ | 6.5 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| | $VAR_2$ | 3 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| | $VAR_3$ | 3 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| | $VAR_4$ | 3 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| | $VAR_5$ | 3 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| | $VAR_6$ | 3 | 4 | 4.5 | 4.5 | 3.5 | 4 |
| **WFG3** | MOEA/D | 4 | 4 | 1 | 1 | 4 | 1 |
| | $VAR_1$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_2$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_3$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_4$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_5$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_6$ | 4 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| **WFG4** | MOEA/D | 6 | 7 | 6 | 1 | 7 | 6 |
| | $VAR_1$ | 6 | 3.5 | 6 | 4.5 | 3.5 | 6 |
| | $VAR_2$ | 2.5 | 3.5 | 1 | 4.5 | 3.5 | 1 |
| | $VAR_3$ | 2.5 | 3.5 | 1 | 4.5 | 3.5 | 1 |
| | $VAR_4$ | 2.5 | 3.5 | 1 | 4.5 | 3.5 | 4 |
| | $VAR_5$ | 2.5 | 3.5 | 4 | 4.5 | 3.5 | 1 |
| | $VAR_6$ | 6 | 3.5 | 6 | 4.5 | 3.5 | 6 |
| **WFG5** | MOEA/D | 6 | 7 | 6 | 1 | 6 | 6 |
| | $VAR_1$ | 6 | 3.5 | 6 | 4.5 | 6 | 6 |
| | $VAR_2$ | 2.5 | 3.5 | 1 | 4.5 | 2.5 | 2.5 |
| | $VAR_3$ | 2.5 | 3.5 | 3.5 | 4.5 | 2.5 | 2.5 |
| | $VAR_4$ | 2.5 | 3.5 | 2 | 4.5 | 2.5 | 2.5 |
| | $VAR_5$ | 2.5 | 3.5 | 3.5 | 4.5 | 2.5 | 2.5 |
| | $VAR_6$ | 6 | 3.5 | 6 | 4.5 | 6 | 6 |
| **WFG6** | MOEA/D | 6.5 | 6.5 | 5 | 4 | 7 | 4.5 |
| | $VAR_1$ | 4 | 4 | 1.5 | 4 | 4.5 | 4.5 |
| | $VAR_2$ | 4 | 4 | 5 | 4 | 4.5 | 4.5 |
| | $VAR_3$ | 1 | 1.5 | 5 | 4 | 1.5 | 4.5 |
| | $VAR_4$ | 4 | 4 | 1.5 | 4 | 4.5 | 1 |
| | $VAR_5$ | 2 | 1.5 | 5 | 4 | 1.5 | 4.5 |
| | $VAR_6$ | 6.5 | 6.5 | 5 | 4 | 4.5 | 4.5 |
| **WFG7** | MOEA/D | 2 | 2 | 2.5 | 1 | 6 | 3 |
| | $VAR_1$ | 2 | 2 | 2.5 | 1 | 2.5 | 3 |
| | $VAR_2$ | 5.5 | 5.5 | 5.5 | 5.5 | 6 | 3 |
| | $VAR_3$ | 5.5 | 5.5 | 5.5 | 5.5 | 2.5 | 6 |
| | $VAR_4$ | 5.5 | 5.5 | 5.5 | 5.5 | 6 | 6 |
| | $VAR_5$ | 5.5 | 5.5 | 5.5 | 5.5 | 2.5 | 6 |
| | $VAR_6$ | 2 | 2 | 1 | 1 | 2.5 | 1 |
| **WFG8** | MOEA/D | 4 | 1 | 4 | 1 | 6 | 7 |
| | $VAR_1$ | 4 | 4.5 | 4 | 4.5 | 1.5 | 3.5 |
| | $VAR_2$ | 4 | 4.5 | 4 | 4.5 | 6 | 3.5 |
| | $VAR_3$ | 4 | 4.5 | 4 | 4.5 | 1.5 | 3.5 |
| | $VAR_4$ | 4 | 4.5 | 4 | 4.5 | 3.5 | 3.5 |
| | $VAR_5$ | 4 | 4.5 | 4 | 4.5 | 3.5 | 3.5 |
| | $VAR_6$ | 4 | 4.5 | 4 | 4.5 | 6 | 3.5 |
| **WFG9** | MOEA/D | 6 | 6.5 | 4 | 1.5 | 6.5 | 4 |
| | $VAR_1$ | 6 | 6.5 | 4 | 5 | 6.5 | 4 |
| | $VAR_2$ | 2 | 3.5 | 4 | 5 | 2 | 4 |
| | $VAR_3$ | 2 | 1 | 4 | 5 | 2 | 4 |
| | $VAR_4$ | 4 | 3.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_5$ | 2 | 3.5 | 4 | 5 | 2 | 4 |
| | $VAR_6$ | 6 | 3.5 | 4 | 1.5 | 4.5 | 4 |

## 5 Experimentation and Results

In this study, a comparative analysis of various configurations of the cooperative coevolutionary algorithm was conducted to solve problems from both the WFG1-9 and DTLZ1-7 families.

Different scenarios were explored, considering the presence of five and ten objectives, as well as the application of division strategies by objectives and variables.

Additionally, divisions involving 2, 4, and 8 species were evaluated, using the solving algorithm MOEA/D with its variants: $VAR_1$, $VAR_2$, $VAR_3$, $VAR_4$, $VAR_5$, and $VAR_6$ [14].

ISSN 2007-9737



**Table 8.** Borda count of CO-MOEA/D with variables division and two species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 87 | 70 | 78 | 39 | 94 | 84 |
| **VAR1** | 70 | 61 | 62.5 | 63 | 61.5 | 61.5 |
| **VAR2** | 64 | 63 | 65 | 65 | 67.5 | 63.5 |
| **VAR3** | 44.5 | 65.5 | 52.5 | 77.5 | 42 | 51 |
| **VAR4** | 59 | 65.5 | 57 | 69 | 56.5 | 58 |
| **VAR5** | 48.5 | 68 | 56.5 | 77.5 | 49 | 55.5 |
| **VAR6** | 75 | 55 | 73.5 | 51 | 71.5 | 68.5 |

**Table 9.** Ranking of problem with five objectives, division of variables and four specie

| PROBLEMS WITH FIVE OBJECTIVES. DIVISION BY VARIABLES AND FOUR SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 6 | 6 | 6 | 5.5 | 6 | 6 |
| | | $VAR_1$ | 4 | 2.5 | 4 | 2 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 2.5 | 6 | 2 | 6 | 6 |
| | | $VAR_3$ | 2 | 6 | 2 | 5.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2 | 2.5 | 2 | 5.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2 | 6 | 2 | 5.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 2.5 | 6 | 2 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 7 | 6 | 4 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 3.5 | 2.5 | 4 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 3.5 | 6 | 4 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 3.5 | 2.5 | 4 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 3.5 | 2.5 | 4 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 3.5 | 2.5 | 4 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 3.5 | 6 | 4 | 6 | 6 |
| | DTLZ3 | MOEA/D | 6 | 4.5 | 6 | 3 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 4.5 | 2.5 | 6 | 4 | 2.5 |
| | | $VAR_2$ | 6 | 4.5 | 6 | 3 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 4.5 | 2.5 | 6 | 2 | 2.5 |
| | | $VAR_4$ | 2.5 | 4.5 | 2.5 | 3 | 2 | 2.5 |
| | | $VAR_5$ | 2.5 | 4.5 | 2.5 | 6 | 2 | 2.5 |
| | | $VAR_6$ | 6 | 1 | 6 | 1 | 6 | 6 |
| | DTLZ4 | MOEA/D | 5 | 5 | 6.5 | 3 | 5 | 5.5 |
| | | $VAR_1$ | 5 | 2 | 3 | 3 | 5 | 1 |
| | | $VAR_2$ | 5 | 5 | 6.5 | 3 | 5 | 5.5 |
| | | $VAR_3$ | 1.5 | 5 | 3 | 6.5 | 1.5 | 1 |
| | | $VAR_4$ | 5 | 1 | 3 | 3 | 5 | 5.5 |
| | | $VAR_5$ | 1.5 | 5 | 3 | 6.5 | 1.5 | 1 |
| | | $VAR_6$ | 5 | 5 | 3 | 3 | 5 | 5.5 |
| | DTLZ5 | MOEA/D | 7 | 1 | 7 | 1 | 5.5 | 6 |
| | | $VAR_1$ | 3.5 | 2.5 | 3.5 | 2 | 2 | 2.5 |
| | | $VAR_2$ | 3.5 | 5.5 | 3.5 | 5 | 2 | 2.5 |
| | | $VAR_3$ | 3.5 | 5.5 | 3.5 | 5 | 5.5 | 6 |
| | | $VAR_4$ | 3.5 | 5.5 | 3.5 | 5 | 5.5 | 2.5 |
| | | $VAR_5$ | 3.5 | 5.5 | 3.5 | 5 | 5.5 | 6 |
| | | $VAR_6$ | 3.5 | 2.5 | 3.5 | 5 | 2 | 2.5 |
| | DTLZ6 | MOEA/D | 7 | 5.5 | 7 | 6 | 6 | 6 |
| | | $VAR_1$ | 4 | 1 | 4 | 1.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 1 | 5.5 | 1 | 3.5 | 1 | 1 |
| | | $VAR_3$ | 4 | 5.5 | 4 | 6 | 6 | 6 |
| | | $VAR_4$ | 4 | 1 | 4 | 1.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 4 | 5.5 | 4 | 6 | 6 | 6 |
| | | $VAR_6$ | 4 | 1 | 4 | 3.5 | 2 | 2 |
| | DTLZ7 | MOEA/D | 4.5 | 5 | 4 | 5 | 4.5 | 5 |
| | | $VAR_1$ | 4.5 | 5 | 4 | 5 | 4.5 | 5 |
| | | $VAR_2$ | 4.5 | 1 | 4 | 1.5 | 4.5 | 1.5 |
| | | $VAR_3$ | 4.5 | 5 | 4 | 5 | 4.5 | 5 |
| | | $VAR_4$ | 1 | 5 | 4 | 5 | 1 | 1.5 |
| | | $VAR_5$ | 4.5 | 5 | 4 | 5 | 4.5 | 5 |
| | | $VAR_6$ | 4.5 | 2 | 4 | 1.5 | 4.5 | 5 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | MOEA/D | 4 | 4 | 7 | 6.5 | 4 | 7 |
| | $VAR_1$ | 4 | 4 | 1 | 1 | 4 | 3.5 |
| | $VAR_2$ | 4 | 4 | 4 | 6.5 | 4 | 3.5 |
| | $VAR_3$ | 4 | 4 | 4 | 3.5 | 4 | 3.5 |
| | $VAR_4$ | 4 | 4 | 4 | 3.5 | 4 | 3.5 |
| | $VAR_5$ | 4 | 4 | 4 | 3.5 | 4 | 3.5 |
| | $VAR_6$ | 4 | 4 | 4 | 3.5 | 4 | 3.5 |
| **WFG2** | MOEA/D | 6.5 | 5 | 1.5 | 1 | 6.5 | 4.5 |
| | $VAR_1$ | 6.5 | 5 | 1.5 | 4.5 | 6.5 | 1 |
| | $VAR_2$ | 3 | 5 | 5 | 4.5 | 3 | 4.5 |
| | $VAR_3$ | 3 | 5 | 5 | 4.5 | 3 | 4.5 |
| | $VAR_4$ | 3 | 1.5 | 5 | 4.5 | 3 | 4.5 |
| | $VAR_5$ | 3 | 1.5 | 5 | 4.5 | 3 | 4.5 |
| | $VAR_6$ | 3 | 5 | 5 | 4.5 | 3 | 4.5 |
| **WFG3** | MOEA/D | 4 | 4 | 4 | 1 | 4 | 1 |
| | $VAR_1$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| | $VAR_2$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| | $VAR_3$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| | $VAR_4$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| | $VAR_5$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| | $VAR_6$ | 4 | 4 | 4 | 4.5 | 4 | 4.5 |
| **WFG4** | MOEA/D | 7 | 7 | 6 | 1 | 7 | 7 |
| | $VAR_1$ | 4 | 3.5 | 2.5 | 4.5 | 4.5 | 3.5 |
| | $VAR_2$ | 1 | 3.5 | 2.5 | 4.5 | 1 | 3.5 |
| | $VAR_3$ | 4 | 3.5 | 6 | 4.5 | 4.5 | 3.5 |
| | $VAR_4$ | 4 | 3.5 | 2.5 | 4.5 | 4.5 | 3.5 |
| | $VAR_5$ | 4 | 3.5 | 2.5 | 4.5 | 2 | 3.5 |
| | $VAR_6$ | 4 | 3.5 | 6 | 4.5 | 4.5 | 3.5 |
| **WFG5** | MOEA/D | 7 | 7 | 1 | 1 | 6.5 | 6 |
| | $VAR_1$ | 5 | 3.5 | 5.5 | 4.5 | 6.5 | 6 |
| | $VAR_2$ | 2 | 3.5 | 1 | 4.5 | 4 | 1.5 |
| | $VAR_3$ | 2 | 3.5 | 5.5 | 4.5 | 1.5 | 3.5 |
| | $VAR_4$ | 5 | 3.5 | 1 | 4.5 | 4 | 3.5 |
| | $VAR_5$ | 2 | 3.5 | 5.5 | 4.5 | 1.5 | 1.5 |
| | $VAR_6$ | 5 | 3.5 | 5.5 | 4.5 | 4 | 6 |
| **WFG6** | MOEA/D | 6.5 | 6.5 | 5 | 1 | 6.5 | 4 |
| | $VAR_1$ | 2.5 | 3 | 5 | 4.5 | 4.5 | 4 |
| | $VAR_2$ | 5 | 6.5 | 1.5 | 4.5 | 4.5 | 4 |
| | $VAR_3$ | 2.5 | 3 | 5 | 4.5 | 2 | 4 |
| | $VAR_4$ | 2.5 | 3 | 1.5 | 4.5 | 2 | 4 |
| | $VAR_5$ | 2.5 | 3 | 5 | 4.5 | 2 | 4 |
| | $VAR_6$ | 6.5 | 3 | 5 | 4.5 | 6.5 | 4 |
| **WFG7** | MOEA/D | 2 | 4 | 5 | 1 | 1 | 4 |
| | $VAR_1$ | 2 | 4 | 1 | 4.5 | 1 | 4 |
| | $VAR_2$ | 5.5 | 4 | 5 | 4.5 | 5.5 | 4 |
| | $VAR_3$ | 5.5 | 4 | 5 | 4.5 | 5.5 | 4 |
| | $VAR_4$ | 5.5 | 4 | 5 | 4.5 | 5.5 | 4 |
| | $VAR_5$ | 5.5 | 4 | 5 | 4.5 | 5.5 | 4 |
| | $VAR_6$ | 2 | 4 | 2 | 4.5 | 1 | 4 |
| **WFG8** | MOEA/D | 4.5 | 1 | 4.5 | 1 | 5.5 | 7 |
| | $VAR_1$ | 4.5 | 4.5 | 4.5 | 4.5 | 5.5 | 3.5 |
| | $VAR_2$ | 4.5 | 4.5 | 4.5 | 4.5 | 5.5 | 3.5 |
| | $VAR_3$ | 4.5 | 4.5 | 4.5 | 4.5 | 2.5 | 3.5 |
| | $VAR_4$ | 4.5 | 4.5 | 4.5 | 4.5 | 2.5 | 3.5 |
| | $VAR_5$ | 1 | 4.5 | 1 | 4.5 | 1 | 3.5 |
| | $VAR_6$ | 4.5 | 4.5 | 4.5 | 4.5 | 5.5 | 3.5 |
| **WFG9** | MOEA/D | 6.5 | 7 | 4 | 1 | 6.5 | 4 |
| | $VAR_1$ | 6.5 | 3.5 | 4 | 4.5 | 6.5 | 4 |
| | $VAR_2$ | 2 | 3.5 | 4 | 4.5 | 1 | 4 |
| | $VAR_3$ | 2 | 3.5 | 4 | 4.5 | 1 | 4 |
| | $VAR_4$ | 4.5 | 3.5 | 4 | 4.5 | 4.5 | 4 |
| | $VAR_5$ | 2 | 3.5 | 4 | 4.5 | 1 | 4 |
| | $VAR_6$ | 4.5 | 3.5 | 4 | 4.5 | 4.5 | 4 |

The statistical comparison for each problem was applied using the Quality indicators Avg-Eucl, Avg-Tcheb, Med-Eucl, Med-Tcheb, Min-Eucl, and Min-Tcheb.

The experimental setups included a limit of 100,000 evaluations, a population of 100 individuals, and a neighborhood size T=10.

ISSN 2007-9737



**Table 10.** Borda count of CO-MOEA/D with variables division and four species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 89.5 | 79.5 | 80.5 | 42 | 86.5 | 85 |
| **VAR1** | 65 | 56 | 52.5 | 60.5 | 67 | 53.5 |
| **VAR2** | 63 | 66 | 64.5 | 64.5 | 63 | 61.5 |
| **VAR3** | 52 | 70 | 64.5 | 77.5 | 52.5 | 60.5 |
| **VAR4** | 57.5 | 54.5 | 53 | 66.5 | 56 | 55.5 |
| **VAR5** | 48.5 | 66.5 | 57.5 | 77.5 | 48.5 | 58.5 |
| **VAR6** | 72.5 | 52.5 | 72.5 | 59.5 | 68.5 | 70.5 |

**Table 11.** Ranking of problem with five objectives, division of variables and eight species

| PROBLEMS WITH FIVE OBJECTIVES. DIVISION BY VARIABLES AND EIGHT SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 6 | 5.5 | 6 | 5.5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 1 | 2.5 | 1 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 1 | 6 | 1 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 5.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 5.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 5.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 1 | 6 | 1 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 6.5 | 6 | 4 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 6.5 | 2.5 | 4 | 4 | 2.5 |
| | | $VAR_2$ | 6 | 3 | 6 | 4 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 3 | 2.5 | 4 | 2 | 2.5 |
| | | $VAR_4$ | 2.5 | 3 | 2.5 | 4 | 2 | 2.5 |
| | | $VAR_5$ | 2.5 | 3 | 2.5 | 4 | 2 | 2.5 |
| | | $VAR_6$ | 6 | 3 | 6 | 4 | 6 | 6 |
| | DTLZ3 | MOEA/D | 6 | 1.5 | 6 | 1.5 | 6 | 6 |
| | | $VAR_1$ | 4 | 5 | 4 | 5 | 3 | 3 |
| | | $VAR_2$ | 6 | 1.5 | 6 | 1.5 | 6 | 6 |
| | | $VAR_3$ | 2 | 5 | 2 | 5 | 3 | 3 |
| | | $VAR_4$ | 2 | 5 | 2 | 5 | 1 | 1 |
| | | $VAR_5$ | 2 | 5 | 2 | 5 | 3 | 3 |
| | | $VAR_6$ | 6 | 5 | 6 | 5 | 6 | 6 |
| | DTLZ4 | MOEA/D | 5 | 3 | 4.5 | 3 | 5.5 | 4 |
| | | $VAR_1$ | 5 | 3 | 4.5 | 3 | 5.5 | 4 |
| | | $VAR_2$ | 5 | 3 | 4.5 | 3 | 5.5 | 4 |
| | | $VAR_3$ | 1.5 | 6.5 | 4.5 | 6.5 | 1 | 4 |
| | | $VAR_4$ | 5 | 3 | 1 | 3 | 5.5 | 4 |
| | | $VAR_5$ | 1.5 | 6.5 | 4.5 | 6.5 | 1 | 4 |
| | | $VAR_6$ | 5 | 3 | 4.5 | 3 | 1 | 4 |
| | DTLZ5 | MOEA/D | 4.5 | 1 | 6.5 | 1 | 4.5 | 5.5 |
| | | $VAR_1$ | 1 | 2 | 1 | 4.5 | 4.5 | 2.5 |
| | | $VAR_2$ | 4.5 | 5 | 3.5 | 4.5 | 4.5 | 1 |
| | | $VAR_3$ | 4.5 | 5 | 3.5 | 4.5 | 4.5 | 5.5 |
| | | $VAR_4$ | 4.5 | 5 | 3.5 | 4.5 | 4.5 | 5.5 |
| | | $VAR_5$ | 4.5 | 5 | 3.5 | 4.5 | 4.5 | 5.5 |
| | | $VAR_6$ | 4.5 | 5 | 6.5 | 4.5 | 1 | 2.5 |
| | DTLZ6 | MOEA/D | 5.5 | 6 | 5.5 | 6 | 5 | 5 |
| | | $VAR_1$ | 5.5 | 2 | 5.5 | 2 | 5 | 5 |
| | | $VAR_2$ | 1 | 2 | 1 | 2 | 1.5 | 1 |
| | | $VAR_3$ | 5.5 | 6 | 5.5 | 6 | 5 | 5 |
| | | $VAR_4$ | 1 | 4 | 1 | 4 | 5 | 5 |
| | | $VAR_5$ | 5.5 | 6 | 5.5 | 6 | 5 | 5 |
| | | $VAR_6$ | 1 | 2 | 1 | 2 | 1.5 | 2 |
| | DTLZ7 | MOEA/D | 4 | 2.5 | 4.5 | 1 | 4 | 4.5 |
| | | $VAR_1$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4.5 |
| | | $VAR_2$ | 4 | 1 | 1 | 1 | 4 | 1 |
| | | $VAR_3$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4.5 |
| | | $VAR_4$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4.5 |
| | | $VAR_5$ | 4 | 5.5 | 4.5 | 5.5 | 4 | 4.5 |
| | | $VAR_6$ | 4 | 2.5 | 4.5 | 1 | 4 | 4.5 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | MOEA/D | 4 | 4 | 7 | 7 | 4 | 7 |
| | $VAR_1$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_2$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_3$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_4$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_5$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_6$ | 4 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| **WFG2** | MOEA/D | 5.5 | 4 | 4 | 1 | 4.5 | 4 |
| | $VAR_1$ | 5.5 | 4 | 4 | 4.5 | 4.5 | 4 |
| | $VAR_2$ | 5.5 | 4 | 4 | 4.5 | 4.5 | 4 |
| | $VAR_3$ | 2 | 4 | 4 | 4.5 | 4.5 | 4 |
| | $VAR_4$ | 2 | 4 | 4 | 4.5 | 1 | 4 |
| | $VAR_5$ | 2 | 4 | 4 | 4.5 | 4.5 | 4 |
| | $VAR_6$ | 5.5 | 4 | 4 | 4.5 | 4.5 | 4 |
| **WFG3** | MOEA/D | 5 | 4 | 1 | 1 | 4 | 1 |
| | $VAR_1$ | 5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_2$ | 5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_3$ | 1.5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_4$ | 1.5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_5$ | 5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| | $VAR_6$ | 5 | 4 | 4.5 | 4.5 | 4 | 4.5 |
| **WFG4** | MOEA/D | 7 | 7 | 4 | 1 | 6.5 | 4.5 |
| | $VAR_1$ | 4.5 | 3.5 | 4 | 4.5 | 6.5 | 4.5 |
| | $VAR_2$ | 1.5 | 3.5 | 4 | 4.5 | 2 | 4.5 |
| | $VAR_3$ | 1.5 | 3.5 | 4 | 4.5 | 2 | 4.5 |
| | $VAR_4$ | 4.5 | 3.5 | 4 | 4.5 | 4.5 | 4.5 |
| | $VAR_5$ | 4.5 | 3.5 | 4 | 4.5 | 2 | 4.5 |
| | $VAR_6$ | 4.5 | 3.5 | 4 | 4.5 | 4.5 | 1 |
| **WFG5** | MOEA/D | 7 | 7 | 4 | 1 | 7 | 5.5 |
| | $VAR_1$ | 4 | 3.5 | 4 | 4.5 | 3.5 | 5.5 |
| | $VAR_2$ | 1 | 3.5 | 4 | 4.5 | 3.5 | 1 |
| | $VAR_3$ | 4 | 3.5 | 4 | 4.5 | 3.5 | 2.5 |
| | $VAR_4$ | 4 | 3.5 | 4 | 4.5 | 3.5 | 2.5 |
| | $VAR_5$ | 4 | 3.5 | 4 | 4.5 | 3.5 | 5.5 |
| | $VAR_6$ | 4 | 3.5 | 4 | 4.5 | 3.5 | 5.5 |
| **WFG6** | MOEA/D | 7 | 6 | 4 | 4 | 7 | 4 |
| | $VAR_1$ | 3.5 | 2.5 | 4 | 4 | 3.5 | 4 |
| | $VAR_2$ | 3.5 | 2.5 | 4 | 4 | 3.5 | 4 |
| | $VAR_3$ | 3.5 | 2.5 | 4 | 4 | 3.5 | 4 |
| | $VAR_4$ | 3.5 | 6 | 4 | 4 | 3.5 | 4 |
| | $VAR_5$ | 3.5 | 2.5 | 4 | 4 | 3.5 | 4 |
| | $VAR_6$ | 3.5 | 6 | 4 | 4 | 3.5 | 4 |
| **WFG7** | MOEA/D | 2 | 1 | 4 | 1 | 1 | 4 |
| | $VAR_1$ | 2 | 4.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_2$ | 5.5 | 4.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_3$ | 5.5 | 4.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_4$ | 5.5 | 4.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_5$ | 5.5 | 4.5 | 4 | 5 | 4.5 | 4 |
| | $VAR_6$ | 2 | 4.5 | 4 | 2 | 4.5 | 4 |
| **WFG8** | MOEA/D | 4 | 1 | 4 | 1 | 4 | 4 |
| | $VAR_1$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | $VAR_2$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | $VAR_3$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | $VAR_4$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | $VAR_5$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | $VAR_6$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| **WFG9** | MOEA/D | 6.5 | 6.5 | 4 | 4 | 7 | 4 |
| | $VAR_1$ | 4.5 | 3 | 4 | 4 | 5 | 4 |
| | $VAR_2$ | 1 | 3 | 4 | 4 | 1 | 4 |
| | $VAR_3$ | 2.5 | 3 | 4 | 4 | 1 | 4 |
| | $VAR_4$ | 4.5 | 3 | 4 | 4 | 5 | 4 |
| | $VAR_5$ | 2.5 | 3 | 4 | 4 | 1 | 4 |
| | $VAR_6$ | 6.5 | 6.5 | 4 | 4 | 5 | 4 |

The polynomial mutation probability was set to 0.14, while the SBX crossover probability was kept at 1.0.

Two division strategies were evaluated: one based on objectives (with 5 and 10 objectives) and another by variables (with 9 and 100 variables), considering the division of species (2, 4, and 8).

The Borda Count method was applied to evaluate the results of thirty-five coevolutionaries (see Table 1).

ISSN 2007-9737



**Table 12.** Borda count of CO-MOEA/D with variables division and eight species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 85 | 66.5 | 75 | 43 | 82 | 75 |
| **VAR1** | 61.5 | 58.5 | 60.5 | 64 | 68 | 62 |
| **VAR2** | 63.5 | 50 | 64 | 56 | 64.5 | 58.5 |
| **VAR3** | 51 | 70 | 61 | 76 | 53 | 62 |
| **VAR4** | 55 | 68 | 53 | 70.5 | 58.5 | 60 |
| **VAR5** | 57.5 | 70 | 61 | 76 | 53 | 65 |
| **VAR6** | 71.5 | 62 | 70.5 | 56.5 | 63 | 65.5 |

**Table 13.** Borda count of CO-MOEA/D with objective division and two species

| | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| PROBLEMS WITH TEN OBJECTIVES. DIVISION BY OBJECTIVES AND TWO SPECIES | DTLZ1 | MOEA/D | 3.5 | 6 | 3.5 | 5.5 | 4.5 | 4.5 |
| | | $VAR_1$ | 3.5 | 2.5 | 3.5 | 2 | 1 | 1 |
| | | $VAR_2$ | 3.5 | 6 | 3.5 | 5.5 | 6.5 | 6.5 |
| | | $VAR_3$ | 3.5 | 2.5 | 3.5 | 2 | 1 | 1 |
| | | $VAR_4$ | 3.5 | 2.5 | 3.5 | 2 | 1 | 1 |
| | | $VAR_5$ | 7 | 6 | 7 | 5.5 | 6.5 | 6.5 |
| | | $VAR_6$ | 3.5 | 2.5 | 3.5 | 5.5 | 4.5 | 4.5 |
| | DTLZ2 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ3 | MOEA/D | 1 | 6 | 1 | 6 | 4 | 4 |
| | | $VAR_1$ | 4.5 | 3.5 | 4.5 | 2.5 | 4 | 4 |
| | | $VAR_2$ | 4.5 | 6 | 4.5 | 6 | 4 | 4 |
| | | $VAR_3$ | 4.5 | 3.5 | 4.5 | 2.5 | 4 | 4 |
| | | $VAR_4$ | 4.5 | 1.5 | 4.5 | 2.5 | 4 | 4 |
| | | $VAR_5$ | 4.5 | 1.5 | 4.5 | 2.5 | 4 | 4 |
| | | $VAR_6$ | 4.5 | 6 | 4.5 | 6 | 4 | 4 |
| | DTLZ4 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ5 | MOEA/D | 6 | 6 | 6 | 2 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 2 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 2 | 6 | 6 |
| | DTLZ6 | MOEA/D | 5 | 5 | 5 | 1 | 5 | 5 |
| | | $VAR_1$ | 5 | 5 | 5 | 4.5 | 5 | 5 |
| | | $VAR_2$ | 1.5 | 5 | 1.5 | 4.5 | 1.5 | 1.5 |
| | | $VAR_3$ | 5 | 1.5 | 5 | 4.5 | 5 | 5 |
| | | $VAR_4$ | 5 | 1.5 | 5 | 4.5 | 5 | 5 |
| | | $VAR_5$ | 5 | 5 | 5 | 4.5 | 5 | 5 |
| | | $VAR_6$ | 1.5 | 5 | 1.5 | 4.5 | 1.5 | 1.5 |
| | DTLZ7 | MOEA/D | 6 | 6 | 6 | 6 | 5.5 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2 | 2 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 5.5 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 5.5 | 4 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2 | 2 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2 | 2 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 5.5 | 6 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | MOEA/D | 3 | 3 | 3 | 2 | 2 | 2.5 |
| | $VAR_1$ | 3 | 3 | 3 | 5.5 | 5.5 | 5.5 |
| | $VAR_2$ | 3 | 3 | 3 | 2 | 2 | 2.5 |
| | $VAR_3$ | 6.5 | 6.5 | 6.5 | 5.5 | 5.5 | 5.5 |
| | $VAR_4$ | 3 | 3 | 3 | 5.5 | 5.5 | 5.5 |
| | $VAR_5$ | 6.5 | 6.5 | 6.5 | 5.5 | 5.5 | 5.5 |
| | $VAR_6$ | 3 | 3 | 3 | 2 | 2 | 1 |
| **WFG2** | MOEA/D | 5 | 5 | 1 | 4 | 3 | 4 |
| | $VAR_1$ | 5 | 5 | 5.5 | 4 | 5.5 | 4 |
| | $VAR_2$ | 1 | 1.5 | 5.5 | 4 | 1 | 4 |
| | $VAR_3$ | 5 | 5 | 1 | 4 | 5.5 | 4 |
| | $VAR_4$ | 5 | 5 | 5.5 | 4 | 5.5 | 4 |
| | $VAR_5$ | 5 | 5 | 1 | 4 | 5.5 | 4 |
| | $VAR_6$ | 2 | 1.5 | 5.5 | 4 | 2 | 4 |
| **WFG3** | MOEA/D | 4 | 4 | 6 | 4 | 1 | 6 |
| | $VAR_1$ | 4 | 4 | 2.5 | 4 | 5.5 | 2.5 |
| | $VAR_2$ | 4 | 4 | 6 | 4 | 2.5 | 6 |
| | $VAR_3$ | 4 | 4 | 2.5 | 4 | 5.5 | 2.5 |
| | $VAR_4$ | 4 | 4 | 2.5 | 4 | 5.5 | 2.5 |
| | $VAR_5$ | 4 | 4 | 2.5 | 4 | 5.5 | 2.5 |
| | $VAR_6$ | 4 | 4 | 6 | 4 | 2.5 | 6 |
| **WFG4** | MOEA/D | 6 | 6 | 6 | 4 | 4 | 6 |
| | $VAR_1$ | 2.5 | 1.5 | 3.5 | 4 | 4 | 2.5 |
| | $VAR_2$ | 2.5 | 6 | 6 | 4 | 4 | 6 |
| | $VAR_3$ | 6 | 3.5 | 1.5 | 4 | 4 | 2.5 |
| | $VAR_4$ | 2.5 | 1.5 | 3.5 | 4 | 4 | 2.5 |
| | $VAR_5$ | 6 | 3.5 | 1.5 | 4 | 4 | 2.5 |
| | $VAR_6$ | 2.5 | 6 | 6 | 4 | 4 | 6 |
| **WFG5** | MOEA/D | 1 | 1 | 4.5 | 7 | 3 | 4.5 |
| | $VAR_1$ | 4.5 | 5.5 | 4.5 | 3.5 | 3 | 1 |
| | $VAR_2$ | 3 | 2.5 | 4.5 | 3.5 | 3 | 4.5 |
| | $VAR_3$ | 6.5 | 5.5 | 4.5 | 3.5 | 6.5 | 4.5 |
| | $VAR_4$ | 4.5 | 5.5 | 4.5 | 3.5 | 3 | 2 |
| | $VAR_5$ | 6.5 | 5.5 | 1 | 3.5 | 6.5 | 7 |
| | $VAR_6$ | 2 | 2.5 | 4.5 | 3.5 | 3 | 4.5 |
| **WFG6** | MOEA/D | 5 | 5 | 5 | 4 | 5.5 | 5.5 |
| | $VAR_1$ | 1.5 | 1 | 5 | 4 | 1 | 2.5 |
| | $VAR_2$ | 5 | 5 | 5 | 4 | 5.5 | 5.5 |
| | $VAR_3$ | 5 | 5 | 1.5 | 4 | 5.5 | 1 |
| | $VAR_4$ | 1.5 | 2 | 5 | 4 | 1 | 5.5 |
| | $VAR_5$ | 5 | 5 | 1.5 | 4 | 1 | 2.5 |
| | $VAR_6$ | 5 | 5 | 5 | 4 | 5.5 | 5.5 |
| **WFG7** | MOEA/D | 5 | 5 | 6 | 6 | 5 | 6 |
| | $VAR_1$ | 2 | 1.5 | 2.5 | 2.5 | 1.5 | 3.5 |
| | $VAR_2$ | 5 | 5 | 6 | 6 | 5 | 6 |
| | $VAR_3$ | 5 | 5 | 2.5 | 2.5 | 5 | 1.5 |
| | $VAR_4$ | 1 | 1.5 | 2.5 | 2.5 | 1.5 | 3.5 |
| | $VAR_5$ | 5 | 5 | 2.5 | 2.5 | 5 | 1.5 |
| | $VAR_6$ | 5 | 5 | 6 | 6 | 5 | 6 |
| **WFG8** | MOEA/D | 5 | 7 | 4.5 | 5 | 5.5 | 2.5 |
| | $VAR_1$ | 5 | 3.5 | 4.5 | 5 | 5.5 | 5.5 |
| | $VAR_2$ | 5 | 3.5 | 4.5 | 5 | 2.5 | 1 |
| | $VAR_3$ | 2 | 3.5 | 4.5 | 1.5 | 2.5 | 5.5 |
| | $VAR_4$ | 5 | 3.5 | 4.5 | 5 | 5.5 | 5.5 |
| | $VAR_5$ | 1 | 3.5 | 1 | 1.5 | 1 | 5.5 |
| | $VAR_6$ | 5 | 3.5 | 4.5 | 5 | 5.5 | 2.5 |
| **WFG9** | MOEA/D | 2 | 4 | 5.5 | 4.5 | 2 | 6 |
| | $VAR_1$ | 5.5 | 4 | 2 | 4.5 | 5.5 | 2 |
| | $VAR_2$ | 2 | 4 | 5.5 | 4.5 | 2 | 6 |
| | $VAR_3$ | 5.5 | 4 | 2 | 4.5 | 5.5 | 2 |
| | $VAR_4$ | 5.5 | 4 | 5.5 | 4.5 | 5.5 | 4 |
| | $VAR_5$ | 5.5 | 4 | 2 | 1 | 5.5 | 2 |
| | $VAR_6$ | 2 | 4 | 5.5 | 4.5 | 2 | 6 |

For each problem, a classification of the seven variants was carried out with five and ten objectives ordered from lowest to highest (see Table 3 a la 22). To calculate the Borda count, the results of each indicator for each problem were added.

ISSN 2007-9737



**Table 14.** Ranking of problem with ten objectives, division of objectives and two species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 69.5 | 81 | 75 | 73 | 68 | 80.5 |
| **VAR1** | 56 | 50 | 56 | 59 | 56.5 | 48.5 |
| **VAR2** | 64 | 75.5 | 79.5 | 73 | 63 | 77.5 |
| **VAR3** | 68.5 | 59.5 | 49.5 | 55.5 | 68.5 | 50.5 |
| **VAR4** | 55 | 45.5 | 59.5 | 59 | 56.5 | 54.5 |
| **VAR5** | 71 | 64.5 | 46 | 55.5 | 64.5 | 58 |
| **VAR6** | 64 | 72 | 79.5 | 73 | 65 | 75.5 |

**Table 15.** Borda count of CO-MOEA/D with objective division and four species

| PROBLEMS WITH TEN OBJECTIVES. DIVISION BY OBJECTIVES AND FOUR SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 2 | 4 | 2 | 4 | 1 | 1 |
| | | $VAR_1$ | 5.5 | 4 | 5.5 | 4 | 5.5 | 5.5 |
| | | $VAR_2$ | 2 | 4 | 2 | 4 | 1 | 1 |
| | | $VAR_3$ | 5.5 | 4 | 5.5 | 4 | 5.5 | 5.5 |
| | | $VAR_4$ | 5.5 | 4 | 5.5 | 4 | 5.5 | 5.5 |
| | | $VAR_5$ | 5.5 | 4 | 5.5 | 4 | 5.5 | 5.5 |
| | | $VAR_6$ | 2 | 4 | 2 | 4 | 1 | 1 |
| | DTLZ2 | MOEA/D | 4 | 5 | 5 | 6 | 4 | 4.5 |
| | | $VAR_1$ | 4 | 5 | 1.5 | 4 | 4 | 4.5 |
| | | $VAR_2$ | 4 | 5 | 5 | 6 | 4 | 4.5 |
| | | $VAR_3$ | 4 | 2 | 5 | 1 | 4 | 4.5 |
| | | $VAR_4$ | 4 | 5 | 1.5 | 1 | 4 | 1 |
| | | $VAR_5$ | 4 | 1 | 5 | 1 | 4 | 4.5 |
| | | $VAR_6$ | 4 | 5 | 5 | 6 | 4 | 4.5 |
| | DTLZ3 | MOEA/D | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_1$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_2$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | | $VAR_3$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_4$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_5$ | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_6$ | 2 | 2 | 2 | 2 | 2 | 2 |
| | DTLZ4 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ5 | MOEA/D | 6 | 6 | 6 | 1 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 4.5 | 1.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 4.5 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 4.5 | 3.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 4.5 | 1.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 4.5 | 3.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 4.5 | 6 | 6 |
| | DTLZ6 | MOEA/D | 4 | 1 | 4 | 1 | 4 | 4 |
| | | $VAR_1$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | | $VAR_2$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | | $VAR_3$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | | $VAR_4$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | | $VAR_5$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | | $VAR_6$ | 4 | 4.5 | 4 | 4.5 | 4 | 4 |
| | DTLZ7 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 4 | 4 | 3.5 | 2.5 | 3 | 3.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2 | 1 | 2 | 2.5 | 3 | 1.5 |
| | | $VAR_4$ | 2 | 1 | 3.5 | 2.5 | 3 | 3.5 |
| | | $VAR_5$ | 2 | 1 | 1 | 2.5 | 1 | 1.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | MOEA/D | 2 | 1 | 2 | 1 | 1 | 3 |
| | $VAR_1$ | 4 | 4 | 4 | 4.5 | 3.5 | 3 |
| | $VAR_2$ | 2 | 2 | 2 | 1 | 3.5 | 3 |
| | $VAR_3$ | 6.5 | 6.5 | 6 | 6.5 | 6.5 | 6.5 |
| | $VAR_4$ | 5 | 4 | 6 | 4.5 | 3.5 | 3 |
| | $VAR_5$ | 6.5 | 6.5 | 6 | 6.5 | 6.5 | 6.5 |
| | $VAR_6$ | 2 | 4 | 2 | 1 | 3.5 | 3 |
| **WFG2** | MOEA/D | 2 | 1 | 1 | 2 | 3 | 4 |
| | $VAR_1$ | 6 | 4 | 5.5 | 5.5 | 3 | 4 |
| | $VAR_2$ | 2 | 4 | 1 | 2 | 3 | 4 |
| | $VAR_3$ | 6 | 6.5 | 5.5 | 5.5 | 6.5 | 4 |
| | $VAR_4$ | 4 | 4 | 5.5 | 5.5 | 3 | 4 |
| | $VAR_5$ | 6 | 6.5 | 5.5 | 5.5 | 6.5 | 4 |
| | $VAR_6$ | 2 | 2 | 1 | 2 | 3 | 4 |
| **WFG3** | MOEA/D | 2 | 2 | 6 | 5 | 2 | 6 |
| | $VAR_1$ | 5.5 | 5.5 | 2.5 | 1.5 | 5.5 | 2.5 |
| | $VAR_2$ | 2 | 2 | 6 | 5 | 2 | 6 |
| | $VAR_3$ | 5.5 | 5.5 | 2.5 | 5 | 5.5 | 2.5 |
| | $VAR_4$ | 5.5 | 5.5 | 2.5 | 1.5 | 5.5 | 2.5 |
| | $VAR_5$ | 5.5 | 5.5 | 2.5 | 5 | 5.5 | 2.5 |
| | $VAR_6$ | 2 | 2 | 6 | 5 | 2 | 6 |
| **WFG4** | MOEA/D | 3 | 2 | 6 | 4 | 1 | 6 |
| | $VAR_1$ | 5.5 | 5.5 | 2 | 4 | 5.5 | 2 |
| | $VAR_2$ | 1 | 2 | 6 | 4 | 1 | 6 |
| | $VAR_3$ | 5.5 | 5.5 | 2 | 4 | 5.5 | 2 |
| | $VAR_4$ | 5.5 | 5.5 | 4 | 4 | 5.5 | 4 |
| | $VAR_5$ | 5.5 | 5.5 | 2 | 4 | 5.5 | 2 |
| | $VAR_6$ | 2 | 2 | 6 | 4 | 1 | 6 |
| **WFG5** | MOEA/D | 2 | 1 | 5.5 | 5 | 1 | 6 |
| | $VAR_1$ | 5.5 | 5.5 | 5.5 | 5 | 5.5 | 3.5 |
| | $VAR_2$ | 2 | 1 | 5.5 | 5 | 2 | 6 |
| | $VAR_3$ | 5.5 | 5.5 | 2 | 1.5 | 5.5 | 2 |
| | $VAR_4$ | 5.5 | 5.5 | 3 | 5 | 5.5 | 3.5 |
| | $VAR_5$ | 5.5 | 5.5 | 1 | 1.5 | 5.5 | 1 |
| | $VAR_6$ | 2 | 1 | 5.5 | 5 | 3 | 6 |
| **WFG6** | MOEA/D | 5 | 5 | 6 | 6 | 5 | 6 |
| | $VAR_1$ | 1 | 1 | 4 | 2 | 1.5 | 6 |
| | $VAR_2$ | 5 | 5 | 6 | 6 | 5 | 6 |
| | $VAR_3$ | 5 | 5 | 2 | 2 | 5 | 1.5 |
| | $VAR_4$ | 2 | 2 | 2 | 4 | 1.5 | 3.5 |
| | $VAR_5$ | 5 | 5 | 2 | 2 | 5 | 1.5 |
| | $VAR_6$ | 5 | 5 | 6 | 6 | 5 | 3.5 |
| **WFG7** | MOEA/D | 2 | 6 | 6 | 6 | 2 | 6 |
| | $VAR_1$ | 5.5 | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 |
| | $VAR_2$ | 2 | 6 | 6 | 6 | 2 | 6 |
| | $VAR_3$ | 5.5 | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 |
| | $VAR_4$ | 5.5 | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 |
| | $VAR_5$ | 5.5 | 2.5 | 2.5 | 2.5 | 5.5 | 2.5 |
| | $VAR_6$ | 2 | 6 | 6 | 6 | 2 | 6 |
| **WFG8** | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| **WFG9** | MOEA/D | 2 | 2 | 6 | 6 | 1 | 7 |
| | $VAR_1$ | 5.5 | 5.5 | 2.5 | 2.5 | 4.5 | 4 |
| | $VAR_2$ | 2 | 2 | 6 | 6 | 4.5 | 4 |
| | $VAR_3$ | 5.5 | 5.5 | 2.5 | 2.5 | 4.5 | 4 |
| | $VAR_4$ | 5.5 | 5.5 | 2.5 | 2.5 | 4.5 | 4 |
| | $VAR_5$ | 5.5 | 5.5 | 2.5 | 2.5 | 4.5 | 1 |
| | $VAR_6$ | 2 | 2 | 6 | 6 | 4.5 | 4 |

For example, in Table 3, the AVG-EUCL results of the MOEA/D variant of the DTLZ1 problem were added with the AVG-EUCL results of the VAR1 variant, for each problem (DTLZ1- WFG9).

This allows for the weighted sums, shown in Table 4. Given that these are minimization problems, the configurations showing the lowest values for each indicator were chosen (highlighted in bold). The purpose of this selection was to determine which of the best configurations stood out as the most outstanding.

This indicates the best parameter configuration for the cooperative coevolutionary algorithm.

ISSN 2007-9737



**Table 16.** Ranking of problem with ten objectives, division of objectives and four species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 56 | 56 | 75.5 | 67 | 51 | 79.5 |
| **VAR1** | 69 | 64 | 56 | 57.5 | 63 | 58 |
| **VAR2** | 54 | 63.5 | 75.5 | 74 | 58 | 76.5 |
| **VAR3** | 73.5 | 67 | 54.5 | 56.5 | 75 | 53.5 |
| **VAR4** | 67 | 62 | 55.5 | 56.5 | 63 | 54 |
| **VAR5** | 73.5 | 66 | 52.5 | 56.5 | 73 | 49.5 |
| **VAR6** | 55 | 63.5 | 75.5 | 74 | 59 | 74 |

**Table 17.** Borda count of CO-MOEA/D with division by variables and two species

| | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| **PROBLEMS WITH TEN OBJECTIVES. DIVISION BY VARIABLES AND TWO SPECIES** | DTLZ1 | MOEA/D | 6 | 5 | 6 | 4.5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2 | 2.5 | 4.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 1 | 6 | 1 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 5 | 2.5 | 4.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 5 | 2.5 | 4.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 5 | 2.5 | 4.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 5 | 6 | 4.5 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 6 | 6 | 6.5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 3.5 | 2.5 | 1.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 4 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 3.5 | 2.5 | 4 | 2.5 | 2.5 |
| | | $VAR_4$ | 4 | 1.5 | 4 | 1.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 1 | 1.5 | 1 | 4 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6.5 | 6 | 6 |
| | DTLZ3 | MOEA/D | 4.5 | 6 | 4.5 | 5 | 4 | 4 |
| | | $VAR_1$ | 4.5 | 2.5 | 4.5 | 5 | 4 | 4 |
| | | $VAR_2$ | 1 | 2.5 | 1 | 1.5 | 4 | 4 |
| | | $VAR_3$ | 4.5 | 6 | 4.5 | 5 | 4 | 4 |
| | | $VAR_4$ | 4.5 | 2.5 | 4.5 | 5 | 4 | 4 |
| | | $VAR_5$ | 4.5 | 6 | 4.5 | 5 | 4 | 4 |
| | | $VAR_6$ | 4.5 | 2.5 | 4.5 | 1.5 | 4 | 4 |
| | DTLZ4 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ5 | MOEA/D | 7 | 7 | 7 | 1 | 7 | 7 |
| | | $VAR_1$ | 4.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 4.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | | $VAR_3$ | 1.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | | $VAR_4$ | 4.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 1.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | | $VAR_6$ | 4.5 | 3.5 | 3.5 | 4.5 | 3.5 | 3.5 |
| | DTLZ6 | MOEA/D | 6 | 5.5 | 6 | 1 | 5.5 | 5.5 |
| | | $VAR_1$ | 6 | 5.5 | 6 | 2.5 | 5.5 | 5.5 |
| | | $VAR_2$ | 1 | 1 | 1 | 5.5 | 1 | 1 |
| | | $VAR_3$ | 2 | 2.5 | 2 | 5.5 | 3 | 2.5 |
| | | $VAR_4$ | 3.5 | 5.5 | 3.5 | 5.5 | 5.5 | 5.5 |
| | | $VAR_5$ | 3.5 | 5.5 | 6 | 5.5 | 5.5 | 5.5 |
| | | $VAR_6$ | 6 | 2.5 | 3.5 | 2.5 | 2 | 2.5 |
| | DTLZ7 | MOEA/D | 6.5 | 6 | 6 | 6.5 | 6.5 | 6 |
| | | $VAR_1$ | 3 | 2.5 | 2.5 | 3 | 3 | 2.5 |
| | | $VAR_2$ | 3 | 6 | 6 | 3 | 6.5 | 6 |
| | | $VAR_3$ | 3 | 2.5 | 2.5 | 3 | 3 | 2.5 |
| | | $VAR_4$ | 3 | 2.5 | 2.5 | 3 | 3 | 2.5 |
| | | $VAR_5$ | 3 | 2.5 | 2.5 | 3 | 3 | 2.5 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **WFG1** | **MOEA/D** | 1 | 4 | 7 | 7 | 1 | 7 |
| | **$VAR_1$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| | **$VAR_2$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| | **$VAR_3$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| | **$VAR_4$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| | **$VAR_5$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| | **$VAR_6$** | 4.5 | 4 | 3.5 | 3.5 | 4.5 | 3.5 |
| WFG2 | MOEA/D | 4.5 | 4 | 1 | 4 | 4 | 4 |
| | $VAR_1$ | 1 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_2$ | 4.5 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_3$ | 4.5 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_4$ | 4.5 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_5$ | 4.5 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_6$ | 4.5 | 4 | 4.5 | 4 | 4 | 4 |
| WFG3 | MOEA/D | 7 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_1$ | 3.5 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_2$ | 3.5 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_3$ | 3.5 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_4$ | 3.5 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_5$ | 3.5 | 4.5 | 4 | 4 | 4 | 4.5 |
| | $VAR_6$ | 3.5 | 1 | 4 | 4 | 4 | 1 |
| WFG4 | MOEA/D | 7 | 7 | 7 | 1 | 4.5 | 4 |
| | $VAR_1$ | 3.5 | 3.5 | 3.5 | 4.5 | 4.5 | 4 |
| | $VAR_2$ | 3.5 | 3.5 | 3.5 | 4.5 | 4.5 | 4 |
| | $VAR_3$ | 3.5 | 3.5 | 3.5 | 4.5 | 4.5 | 4 |
| | $VAR_4$ | 3.5 | 3.5 | 3.5 | 4.5 | 4.5 | 4 |
| | $VAR_5$ | 3.5 | 3.5 | 3.5 | 4.5 | 4.5 | 4 |
| | $VAR_6$ | 3.5 | 3.5 | 3.5 | 4.5 | 1 | 4 |
| WFG5 | MOEA/D | 7 | 7 | 7 | 4 | 4 | 4 |
| | $VAR_1$ | 3.5 | 3.5 | 4.5 | 4 | 4 | 4 |
| | $VAR_2$ | 3.5 | 3.5 | 1.5 | 4 | 4 | 4 |
| | $VAR_3$ | 3.5 | 3.5 | 1.5 | 4 | 4 | 4 |
| | $VAR_4$ | 3.5 | 3.5 | 4.5 | 4 | 4 | 4 |
| | $VAR_5$ | 3.5 | 3.5 | 4.5 | 4 | 4 | 4 |
| | $VAR_6$ | 3.5 | 3.5 | 4.5 | 4 | 4 | 4 |
| WFG6 | MOEA/D | 7 | 6.5 | 4 | 4 | 4.5 | 4 |
| | $VAR_1$ | 3.5 | 3 | 4 | 4 | 4.5 | 4 |
| | $VAR_2$ | 3.5 | 3 | 4 | 4 | 4.5 | 4 |
| | $VAR_3$ | 3.5 | 3 | 4 | 4 | 4.5 | 4 |
| | $VAR_4$ | 3.5 | 3 | 4 | 4 | 4.5 | 4 |
| | $VAR_5$ | 3.5 | 6.5 | 4 | 4 | 1 | 4 |
| | $VAR_6$ | 3.5 | 3 | 4 | 4 | 4.5 | 4 |
| WFG7 | MOEA/D | 7 | 7 | 5.5 | 4 | 6 | 1 |
| | $VAR_1$ | 3.5 | 3.5 | 5.5 | 4 | 2.5 | 5.5 |
| | $VAR_2$ | 3.5 | 3.5 | 2 | 4 | 6 | 2.5 |
| | $VAR_3$ | 3.5 | 3.5 | 5.5 | 4 | 2.5 | 5.5 |
| | $VAR_4$ | 3.5 | 3.5 | 2 | 4 | 2.5 | 5.5 |
| | $VAR_5$ | 3.5 | 3.5 | 2 | 4 | 2.5 | 5.5 |
| | $VAR_6$ | 3.5 | 3.5 | 5.5 | 4 | 6 | 2.5 |
| WFG8 | MOEA/D | 4.5 | 4 | 6 | 4 | 6 | 1 |
| | $VAR_1$ | 4.5 | 4 | 6 | 4 | 2 | 5.5 |
| | $VAR_2$ | 4.5 | 4 | 2.5 | 4 | 6 | 2.5 |
| | $VAR_3$ | 1 | 4 | 2.5 | 4 | 4 | 5.5 |
| | $VAR_4$ | 4.5 | 4 | 6 | 4 | 2 | 5.5 |
| | $VAR_5$ | 4.5 | 4 | 2.5 | 4 | 2 | 5.5 |
| | $VAR_6$ | 4.5 | 4 | 2.5 | 4 | 6 | 2.5 |
| WFG9 | MOEA/D | 4 | 4 | 4.5 | 6 | 4 | 4 |
| | $VAR_1$ | 4 | 4 | 4.5 | 2.5 | 4 | 4 |
| | $VAR_2$ | 4 | 4 | 1 | 2.5 | 4 | 4 |
| | $VAR_3$ | 4 | 4 | 4.5 | 2.5 | 4 | 4 |
| | $VAR_4$ | 4 | 4 | 4.5 | 6 | 4 | 4 |
| | $VAR_5$ | 4 | 4 | 4.5 | 6 | 4 | 4 |
| | $VAR_6$ | 4 | 4 | 4.5 | 2.5 | 4 | 4 |

Based on the results obtained from experimentation, the configurations with the best performance are shown in Fig. 4 for problems with five objectives, and for problems with ten objectives are shown in Fig. 5. It was observed that the variants that provided high-quality results were

ISSN 2007-9737



**Table 18.** Ranking of problem with ten objectives, division of variables and two species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 91 | 89.5 | 87.5 | 68.5 | 79 | 74 |
| **VAR1** | 57 | 56 | 64 | 58 | 57.5 | 62 |
| **VAR2** | 62.5 | 60 | 56 | 60 | 74.5 | 65.5 |
| **VAR3** | 50 | 59.5 | 53.5 | 63.5 | 57 | 59 |
| **VAR4** | 59.5 | 57 | 59.5 | 64.5 | 57.5 | 62 |
| **VAR5** | 53.5 | 64 | 55.5 | 67 | 54 | 62 |
| **VAR6** | 74.5 | 62 | 72 | 66.5 | 68.5 | 63.5 |

**Table 19.** Borda count of CO-MOEA/D with division by variables and four species

| PROBLEMS WITH TEN OBJECTIVES. DIVISION BY VARIABLES AND FOUR SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 6 | 4.5 | 6 | 5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 4.5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 4.5 | 6 | 2 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 4.5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 4.5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 4.5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 1 | 6 | 1 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 6 | 6 | 7 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 1 | 2 | 2 |
| | | $VAR_2$ | 6 | 6 | 6 | 4 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 4 | 2 | 2 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 4 | 4 | 4 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 4 | 2 | 2 |
| | | $VAR_6$ | 6 | 6 | 6 | 4 | 6 | 6 |
| | DTLZ3 | MOEA/D | 6 | 4 | 6 | 4.5 | 6 | 6 |
| | | $VAR_1$ | 3 | 4 | 3 | 4.5 | 1.5 | 1.5 |
| | | $VAR_2$ | 6 | 4 | 6 | 1 | 6 | 6 |
| | | $VAR_3$ | 1 | 4 | 1 | 4.5 | 1.5 | 1.5 |
| | | $VAR_4$ | 3 | 4 | 3 | 4.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 3 | 4 | 3 | 4.5 | 3.5 | 3.5 |
| | | $VAR_6$ | 6 | 4 | 6 | 4.5 | 6 | 6 |
| | DTLZ4 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ5 | MOEA/D | 7 | 7 | 7 | 1 | 7 | 7 |
| | | $VAR_1$ | 4 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 4 | 1 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_3$ | 1 | 4 | 1 | 4.5 | 3.5 | 3.5 |
| | | $VAR_4$ | 4 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 4 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_6$ | 4 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | DTLZ6 | MOEA/D | 4.5 | 5 | 4.5 | 2 | 5 | 5 |
| | | $VAR_1$ | 4.5 | 5 | 4.5 | 2 | 5 | 5 |
| | | $VAR_2$ | 1 | 1 | 1 | 4 | 1 | 1 |
| | | $VAR_3$ | 4.5 | 5 | 4.5 | 6 | 5 | 5 |
| | | $VAR_4$ | 4.5 | 5 | 4.5 | 6 | 5 | 5 |
| | | $VAR_5$ | 4.5 | 5 | 4.5 | 6 | 5 | 5 |
| | | $VAR_6$ | 4.5 | 2 | 4.5 | 2 | 2 | 2 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DTLZ7 | MOEA/D | 7 | 6.5 | 6 | 7 | 6.5 | 6 |
| | $VAR_1$ | 3.5 | 3 | 2.5 | 3.5 | 3 | 2.5 |
| | $VAR_2$ | 3.5 | 3 | 6 | 3.5 | 6.5 | 6 |
| | $VAR_3$ | 3.5 | 3 | 2.5 | 3.5 | 3 | 2.5 |
| | $VAR_4$ | 3.5 | 3 | 2.5 | 3.5 | 3 | 2.5 |
| | $VAR_5$ | 3.5 | 3 | 2.5 | 3.5 | 3 | 2.5 |
| | $VAR_6$ | 3.5 | 6.5 | 6 | 3.5 | 3 | 6 |
| WFG1 | MOEA/D | 1.5 | 4 | 7 | 7 | 4 | 7 |
| | $VAR_1$ | 5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_2$ | 1.5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_3$ | 5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_4$ | 5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_5$ | 5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| | $VAR_6$ | 5 | 4 | 3.5 | 3.5 | 4 | 3.5 |
| WFG2 | MOEA/D | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_1$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_2$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_3$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_4$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_5$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_6$ | 4 | 4 | 4 | 4 | 4 | 4 |
| WFG3 | MOEA/D | 6 | 4 | 4 | 4 | 4 | 5.5 |
| | $VAR_1$ | 6 | 4 | 4 | 4 | 4 | 5.5 |
| | $VAR_2$ | 2.5 | 4 | 4 | 4 | 4 | 2 |
| | $VAR_3$ | 6 | 4 | 4 | 4 | 4 | 2 |
| | $VAR_4$ | 2.5 | 4 | 4 | 4 | 4 | 5.5 |
| | $VAR_5$ | 2.5 | 4 | 4 | 4 | 4 | 5.5 |
| | $VAR_6$ | 2.5 | 4 | 4 | 4 | 4 | 2 |
| WFG4 | MOEA/D | 7 | 7 | 7 | 1 | 6 | 4.5 |
| | $VAR_1$ | 3.5 | 3.5 | 3.5 | 4.5 | 6 | 4.5 |
| | $VAR_2$ | 3.5 | 3.5 | 3.5 | 4.5 | 2.5 | 4.5 |
| | $VAR_3$ | 3.5 | 3.5 | 3.5 | 4.5 | 6 | 4.5 |
| | $VAR_4$ | 3.5 | 3.5 | 3.5 | 4.5 | 2.5 | 4.5 |
| | $VAR_5$ | 3.5 | 3.5 | 3.5 | 4.5 | 2.5 | 4.5 |
| | $VAR_6$ | 3.5 | 3.5 | 3.5 | 4.5 | 2.5 | 1 |
| WFG5 | MOEA/D | 7 | 7 | 7 | 4 | 6 | 4 |
| | $VAR_1$ | 3.5 | 3.5 | 3.5 | 4 | 6 | 4 |
| | $VAR_2$ | 3.5 | 3.5 | 3.5 | 4 | 2.5 | 4 |
| | $VAR_3$ | 3.5 | 3.5 | 3.5 | 4 | 2.5 | 4 |
| | $VAR_4$ | 3.5 | 3.5 | 3.5 | 4 | 6 | 4 |
| | $VAR_5$ | 3.5 | 3.5 | 3.5 | 4 | 2.5 | 4 |
| | $VAR_6$ | 3.5 | 3.5 | 3.5 | 4 | 2.5 | 4 |
| WFG6 | MOEA/D | 7 | 6 | 4 | 4 | 6 | 4.5 |
| | $VAR_1$ | 3.5 | 2.5 | 4 | 4 | 6 | 4.5 |
| | $VAR_2$ | 3.5 | 2.5 | 4 | 4 | 2.5 | 1 |
| | $VAR_3$ | 3.5 | 6 | 4 | 4 | 2.5 | 4.5 |
| | $VAR_4$ | 3.5 | 2.5 | 4 | 4 | 2.5 | 4.5 |
| | $VAR_5$ | 3.5 | 2.5 | 4 | 4 | 2.5 | 4.5 |
| | $VAR_6$ | 3.5 | 6 | 4 | 4 | 6 | 4.5 |
| WFG7 | MOEA/D | 7 | 7 | 4 | 4 | 6 | 1 |
| | $VAR_1$ | 3.5 | 3.5 | 4 | 4 | 3.5 | 4.5 |
| | $VAR_2$ | 3.5 | 3.5 | 4 | 4 | 6 | 4.5 |
| | $VAR_3$ | 3.5 | 3.5 | 4 | 4 | 1.5 | 4.5 |
| | $VAR_4$ | 3.5 | 3.5 | 4 | 4 | 3.5 | 4.5 |
| | $VAR_5$ | 3.5 | 3.5 | 4 | 4 | 1.5 | 4.5 |
| | $VAR_6$ | 3.5 | 3.5 | 4 | 4 | 6 | 4.5 |
| WFG8 | MOEA/D | 5.5 | 4 | 5.5 | 4 | 5 | 4 |
| | $VAR_1$ | 2.5 | 4 | 5.5 | 4 | 5 | 4 |
| | $VAR_2$ | 5.5 | 4 | 2 | 4 | 5 | 4 |
| | $VAR_3$ | 1 | 4 | 2 | 4 | 1 | 4 |
| | $VAR_4$ | 5.5 | 4 | 2 | 4 | 5 | 4 |
| | $VAR_5$ | 2.5 | 4 | 5.5 | 4 | 2 | 4 |
| | $VAR_6$ | 5.5 | 4 | 5.5 | 4 | 5 | 4 |
| WFG9 | MOEA/D | 4 | 4 | 5 | 4.5 | 4 | 4 |
| | $VAR_1$ | 4 | 4 | 5 | 4.5 | 4 | 4 |
| | $VAR_2$ | 4 | 4 | 5 | 1 | 4 | 4 |
| | $VAR_3$ | 4 | 4 | 1.5 | 4.5 | 4 | 4 |
| | $VAR_4$ | 4 | 4 | 5 | 4.5 | 4 | 4 |
| | $VAR_5$ | 4 | 4 | 1.5 | 4.5 | 4 | 4 |
| | $VAR_6$ | 4 | 4 | 5 | 4.5 | 4 | 4 |

$VAR_2$ and $VAR_6$, both implemented with the variable-based splitting strategy. In the case of problems with ten objectives, it was observed that the $VAR_6$ variant, with division by objectives, showed the best performance (Table 2). Furthermore, the $VAR_4$ variant, with variable-based splitting, also demonstrated outstanding performance.

ISSN 2007-9737



**Table 20**. Ranking of problem with ten objectives, division of variables and four species

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 94 | 89 | 89 | 71.5 | 85 | 82 |
| **VAR1** | 59.5 | 63 | 59 | 62 | 58 | 62 |
| **VAR2** | 62.5 | 51 | 69 | 60 | 65.5 | 56 |
| **VAR3** | 56 | 64.5 | 52.5 | 64.5 | 55 | 58.5 |
| **VAR4** | 53.5 | 61 | 49.5 | 64.5 | 52 | 55.5 |
| **VAR5** | 56 | 64.5 | 62 | 64.5 | 58.5 | 65 |
| **VAR6** | 66.5 | 55 | 67 | 61 | 71 | 66 |

**Table 21.** Borda count of CO-MOEA/D with division by variables and eight species

| PROBLEMS WITH TEN OBJECTIVES. DIVISION BY VARIABLES AND EIGHT SPECIES | PROBLEM | VARIANT | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|---|---|
| | DTLZ1 | MOEA/D | 6 | 5 | 6 | 5 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 1.5 | 6 | 1.5 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 5 | 2.5 | 5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 1.5 | 6 | 1.5 | 6 | 6 |
| | DTLZ2 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 4 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ3 | MOEA/D | 6.5 | 5 | 6.5 | 5 | 7 | 7 |
| | | $VAR_1$ | 3.5 | 5 | 3.5 | 5 | 3.5 | 3.5 |
| | | $VAR_2$ | 3.5 | 1.5 | 3.5 | 1.5 | 3.5 | 3.5 |
| | | $VAR_3$ | 3.5 | 5 | 3.5 | 5 | 3.5 | 3.5 |
| | | $VAR_4$ | 1 | 5 | 1 | 5 | 3.5 | 3.5 |
| | | $VAR_5$ | 3.5 | 5 | 3.5 | 5 | 3.5 | 3.5 |
| | | $VAR_6$ | 6.5 | 1.5 | 6.5 | 1.5 | 3.5 | 3.5 |
| | DTLZ4 | MOEA/D | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_1$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | | $VAR_3$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 6 | 6 | 6 | 6 | 6 | 6 |
| | DTLZ5 | MOEA/D | 7 | 7 | 7 | 1 | 7 | 7 |
| | | $VAR_1$ | 3.5 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_2$ | 3.5 | 1 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_3$ | 3.5 | 4 | 1 | 4.5 | 3.5 | 3.5 |
| | | $VAR_4$ | 3.5 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_5$ | 3.5 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | | $VAR_6$ | 3.5 | 4 | 4 | 4.5 | 3.5 | 3.5 |
| | DTLZ6 | MOEA/D | 5 | 5 | 5 | 5.5 | 3 | 3 |
| | | $VAR_1$ | 5 | 5 | 5 | 3 | 6 | 6 |
| | | $VAR_2$ | 1 | 1 | 1 | 1 | 1 | 1 |
| | | $VAR_3$ | 5 | 5 | 5 | 5.5 | 6 | 6 |
| | | $VAR_4$ | 5 | 5 | 5 | 5.5 | 3 | 3 |
| | | $VAR_5$ | 5 | 5 | 5 | 5.5 | 6 | 6 |
| | | $VAR_6$ | 2 | 2 | 2 | 2 | 3 | 3 |
| | DTLZ7 | MOEA/D | 7 | 7 | 6.5 | 7 | 6 | 6 |
| | | $VAR_1$ | 3.5 | 3.5 | 3 | 3.5 | 2.5 | 2.5 |
| | | $VAR_2$ | 3.5 | 3.5 | 6.5 | 3.5 | 6 | 6 |
| | | $VAR_3$ | 3.5 | 3.5 | 3 | 3.5 | 2.5 | 2.5 |
| | | $VAR_4$ | 3.5 | 3.5 | 3 | 3.5 | 2.5 | 2.5 |
| | | $VAR_5$ | 3.5 | 3.5 | 3 | 3.5 | 2.5 | 2.5 |
| | | $VAR_6$ | 3.5 | 3.5 | 3 | 3.5 | 6 | 6 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| WFG1 | MOEA/D | 2 | 5 | 6.5 | 7 | 2 | 6.5 |
| | $VAR_1$ | 5.5 | 5 | 3.5 | 3.5 | 5.5 | 3.5 |
| | $VAR_2$ | 2 | 1.5 | 3.5 | 3.5 | 2 | 3.5 |
| | $VAR_3$ | 5.5 | 5 | 3.5 | 3.5 | 5.5 | 3.5 |
| | $VAR_4$ | 5.5 | 5 | 3.5 | 3.5 | 5.5 | 3.5 |
| | $VAR_5$ | 5.5 | 5 | 6.5 | 3.5 | 5.5 | 6.5 |
| | $VAR_6$ | 2 | 1.5 | 1 | 3.5 | 2 | 1 |
| WFG2 | MOEA/D | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_1$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_2$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_3$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_4$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_5$ | 4 | 4 | 4 | 4 | 4 | 4 |
| | $VAR_6$ | 4 | 4 | 4 | 4 | 4 | 4 |
| WFG3 | MOEA/D | 7 | 4 | 6 | 4 | 5.5 | 7 |
| | $VAR_1$ | 3.5 | 4 | 2.5 | 4 | 2 | 3.5 |
| | $VAR_2$ | 3.5 | 4 | 6 | 4 | 2 | 3.5 |
| | $VAR_3$ | 3.5 | 4 | 2.5 | 4 | 5.5 | 3.5 |
| | $VAR_4$ | 3.5 | 4 | 6 | 4 | 2 | 3.5 |
| | $VAR_5$ | 3.5 | 4 | 2.5 | 4 | 5.5 | 3.5 |
| | $VAR_6$ | 3.5 | 4 | 2.5 | 4 | 5.5 | 3.5 |
| WFG4 | MOEA/D | 7 | 7 | 6 | 1 | 6.5 | 4.5 |
| | $VAR_1$ | 3.5 | 3.5 | 2.5 | 4.5 | 3 | 4.5 |
| | $VAR_2$ | 3.5 | 3.5 | 2.5 | 4.5 | 3 | 1 |
| | $VAR_3$ | 3.5 | 3.5 | 6 | 4.5 | 3 | 4.5 |
| | $VAR_4$ | 3.5 | 3.5 | 2.5 | 4.5 | 3 | 4.5 |
| | $VAR_5$ | 3.5 | 3.5 | 6 | 4.5 | 6.5 | 4.5 |
| | $VAR_6$ | 3.5 | 3.5 | 2.5 | 4.5 | 3 | 4.5 |
| WFG5 | MOEA/D | 7 | 6.5 | 5 | 4 | 4 | 4.5 |
| | $VAR_1$ | 3.5 | 3 | 5 | 4 | 4 | 4.5 |
| | $VAR_2$ | 3.5 | 3 | 5 | 4 | 4 | 1 |
| | $VAR_3$ | 3.5 | 3 | 1.5 | 4 | 4 | 4.5 |
| | $VAR_4$ | 3.5 | 3 | 1.5 | 4 | 4 | 4.5 |
| | $VAR_5$ | 3.5 | 6.5 | 5 | 4 | 4 | 4.5 |
| | $VAR_6$ | 3.5 | 3 | 5 | 4 | 4 | 4.5 |
| WFG6 | MOEA/D | 7 | 6.5 | 6 | 4 | 7 | 5.5 |
| | $VAR_1$ | 3.5 | 3 | 6 | 4 | 3.5 | 5.5 |
| | $VAR_2$ | 3.5 | 3 | 2.5 | 4 | 3.5 | 2 |
| | $VAR_3$ | 3.5 | 6.5 | 2.5 | 4 | 3.5 | 2 |
| | $VAR_4$ | 3.5 | 3 | 2.5 | 4 | 3.5 | 2 |
| | $VAR_5$ | 3.5 | 3 | 2.5 | 4 | 3.5 | 5.5 |
| | $VAR_6$ | 3.5 | 3 | 6 | 4 | 3.5 | 5.5 |
| WFG7 | MOEA/D | 7 | 7 | 4 | 4 | 6 | 1 |
| | $VAR_1$ | 3.5 | 3.5 | 4 | 4 | 4 | 5.5 |
| | $VAR_2$ | 3.5 | 3.5 | 4 | 4 | 6 | 1 |
| | $VAR_3$ | 3.5 | 3.5 | 4 | 4 | 1 | 5.5 |
| | $VAR_4$ | 3.5 | 3.5 | 4 | 4 | 1 | 5.5 |
| | $VAR_5$ | 3.5 | 3.5 | 4 | 4 | 1 | 5.5 |
| | $VAR_6$ | 3.5 | 3.5 | 4 | 4 | 6 | 1 |
| WFG8 | MOEA/D | 5.5 | 4 | 4 | 4 | 5 | 4 |
| | $VAR_1$ | 5.5 | 4 | 4 | 4 | 5 | 4 |
| | $VAR_2$ | 5.5 | 4 | 4 | 4 | 5 | 4 |
| | $VAR_3$ | 2 | 4 | 4 | 4 | 1.5 | 4 |
| | $VAR_4$ | 2 | 4 | 4 | 4 | 5 | 4 |
| | $VAR_5$ | 2 | 4 | 4 | 4 | 1.5 | 4 |
| | $VAR_6$ | 5.5 | 4 | 4 | 4 | 5 | 4 |
| WFG9 | MOEA/D | 4 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_1$ | 4 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_2$ | 4 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_3$ | 4 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_4$ | 4 | 4 | 1 | 4 | 4 | 4 |
| | $VAR_5$ | 4 | 4 | 4.5 | 4 | 4 | 4 |
| | $VAR_6$ | 4 | 4 | 4.5 | 4 | 4 | 4 |

Regarding the number of species used in these configurations, it was observed that working with 4 species had a significant impact on the quality of the solutions generated for the problems implemented.

## 6 Conclusions and Future Work

It can be concluded that the objective set in this work was successfully achieved through the conduct of an experimental analysis. To achieve this goal, a variety of state-of-the-art benchmark

**Table 22**. Ranking of problem with ten objectives, division of variables and eight specie

| VARIANTE | AVG_EUCL | MIN_EUCL | AVG_TCH | MIN_TCHE | MED_EUCL | MED_TCHE |
|---|---|---|---|---|---|---|
| **MOEA/D** | 94 | 89 | 89 | 71.5 | 85 | 82 |
| **VAR1** | 59.5 | 63 | 59 | 62 | 58 | 62 |
| **VAR2** | 62.5 | 51 | 69 | 60 | 65.5 | 56 |
| **VAR3** | 56 | 64.5 | 52.5 | 64.5 | 55 | 58.5 |
| **VAR4** | 53.5 | 61 | 49.5 | 64.5 | 52 | 55.5 |
| **VAR5** | 56 | 64.5 | 62 | 64.5 | 58.5 | 65 |
| **VAR6** | 66.5 | 55 | 67 | 61 | 71 | 66 |

problems were employed, providing a solid foundation for evaluating the proposed solutions. Configurations were evaluated with 2, 4, and 8 species, aiming to minimize the distance of the non-dominated set concerning the decision-maker's point of interest.

During the experimentation, parameter tuning tests were conducted on the coevolutionary algorithm, both with variable division and with objective division, allowing for the evaluation of different configurations to address multi-objective problems. It was validated that there is no single configuration that provides good solutions, but rather there are various options that can lead to viable solutions that approach the decision-maker's region of interest.

It was observed that variable division and four species predominated. Additionally, it was highlighted that the $VAR_6$ variant excels for both five and ten objectives. Therefore, it could be considered that $VAR_6$, along with a strategy of objective or variable division and 4 species, maybe a viable configuration that offers good results.

As a future work, it would be interesting to expand this study into a dynamic environment, where the parameters and/or conditions of the problem may change over time. This would allow evaluation of how the adaptability of the cooperative coevolutionary algorithm impacts the quality of the solutions generated concerning the DM's region of interest. Such an approach could provide a more comprehensive understanding of the algorithm's effectiveness in real-world situations where conditions may change unexpectedly.

## Acknowledgments

The authors want to thank Laboratorio Nacional de Tecnologías de la Información and the support of: (a) Cátedras CONAHCYT Program Number 3058, (b) the TecNM project 21336.24-P, (c) the support granted through the Scholarship for Postgraduate Studies with CVU 960719.

## References

1. **Kozioł-Nadolna, K., Beyer, K. (2021).** Determinants of the decision-making process in organizations. Procedia Computer Science**,** Vol. 192. pp. 2375–2384. DOI: 10.1016/j.procs.2021.09.006.
2. **Li, G., Wang, G. G., Wang, S. (2021).** Two-population coevolutionary algorithm with dynamic learning strategy for many-objective optimization. Mathematics, Vol. 9, No 4, p. 420. DOI: 10.3390/math9040420.
3. **Zhong, R., Munetomo, M. (2022).** Cooperative coevolutionary hybrid NSGA-II with linkage measurement minimization for large-scale multi-objective optimization. arXiv preprintarXiv:2208.13415**.** DOI: 10.48550/arXiv.2208.13415.
4. **Navas, M. M., Urbaneja, A. J. N. (2013).** Metaheurísticas multiobjetivo adaptativas. Computación y Sistemas, Vol. 17 No.1, pp. 53–62 ISSN 1405-55465.
5. **Potter, M. A., Jong, K. A. D. (2000).** Cooperative coevolution: An architecture for evolving coadapted subcomponents.

Evolutionary computation, Vol. 8. pp. 1–29. DOI: 10.1162/106365600568086.

6. **Ríos-Insua, S., Mateos-Caballero, A., Jiménez-Martín, A. (2004).** Optimización multiobjetivo basada en metaheurísticas. Anales de la Real Academia de Doctores de España, Real Academia de Doctores.Vol. 8, pp. 159–177.
7. **Rivera, G., Florencia, R., Guerrero, M., Porras, R., Sánchez-Solís, J. P. (2021)** Online multi-criteria portfolio analysis through compromise programming models built on the underlying principles of fuzzy outranking. Information Sciences, Vol. 580, pp. 734–755. DOI: 10.1016/j.ins.2021.08.087.
8. **Rivera, G., Porras, R., Sanchez-Solis, J. P., Florencia, R., García, V. (2022)** Outranking-based multi-objective PSO for scheduling unrelated parallel machines with a freight industry-oriented application. Engineering Applications of Artificial Intelligence, Vol. 108, p. 104556. DOI: 10.1016/j.engappai.2021. 104556.
9. **Darwin's, C. (1859).** On the origin of species. Published on, Vol. 24, No. 1.
10. **De-Jong, K. A. (1975).** An analysis of the behavior of a class of genetic adaptive systems. University of Michigan.
11. **Gong, D., Xu, B., Zhang, Y., Guo, Y., Yang, S. (2019).** A similarity-based cooperative co-evolutionary algorithm for dynamic interval multiobjective optimization problems. IEEE Transactions on Evolutionary Computation, Vol. 24, No. 1, pp. 142–156. DOI: 10.1109/ TEVC.2019.2912204.
12. **Xu, B., Gong, D., Zhang, Y., Yang, S., Wang, L., Fan, Z., Zhang, Y. (2022).** Cooperative co-evolutionary algorithm for multi-objective optimization problems with changing decision variables. Information Sciences, Vol. 607, pp. 278–296. DOI: 10.1016/j.ins.2022.05.12311.
13. **Fernandez, E., Lopez, E., Coello, C., Navarro, J. (2010).** Evolutionary multiobjective optimization using an outranking-based dominance generalization. Computers & Operations Research. Vol. 37. pp. 390–395. DOI: 10.1016/j.cor.2009.06.004.
14. **Fernandez, E., Rangel-Valdez, N., Cruz-Reyes, L., Gomez-Santillan, C., Coello-Coello, C. A. (2021).** Preference incorporation into MOEA/D using an outranking approach with imprecise model parameters. Social Science Research Network, 2021, pp. 1–24. DOI: 10.2139/ssrn.3960041.
15. **Mejía-de-Dios, J. A., Mezura-Montes, E., Quiroz, M. (2021).** Automated parameter tuning as a bilevel optimization problem solved by a surrogate-assisted population-based approach. Applied Intelligence, Vol. 51, pp. 1–23. DOI: 10.1007/s10489-020-02151-y.
16. **Romero-Ocaño. A., Cosío-León, M., Valenzuela-Alcaraz, V., Avilés-Rodríguez, G. Martínez-Vargas, A. (2018).** Effect of parameters tuned by a taguchi design L_9 3^4 in the GRASP algorithm to solve the vehicle routing problem with time windows. Computación y Sistemas, Vol. 22, No. 2, pp. 657–673. DOI: 10.13053/CyS-22-2-2595.
17. **Castellanos, A.; Cruz-Reyes, L.; Fernández, E.; Rivera, G.; Gomez-Santillan, C.; Rangel-Valdez, N. (2022).** Hybridisation of swarm intelligence algorithms with multi-criteria ordinal classification: A strategy to address many-objective optimisation. Mathematics, Vol. 10, p. 322. DOI: 10.3390/math10030322.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Claudia Guadalupe Gómez Santillán.*

ISSN 2007-9737

# Framework to Support Radiologist Personnel in the Diagnosis of Diseases in Medical Images Using Deep Learning and Personalized DICOM Tags

Manuel Rodriguez-Contreras, J. Patricia Sánchez-Solís* , Gilberto Rivera, Rogelio Florencia

Universidad Autónoma de Ciudad Juárez,
División Multidisciplinaria de Ciudad Universitaria,
Mexico

manuel.manny45@gmail.com, {julia.sanchez, gilberto.rivera, rogelio.florencia}@uacj.mx

**Abstract.** Technological innovations in the healthcare field have allowed medical images to be widely used in the diagnostic care of patients since medical personnel can analyze different body organs to identify any disease through these images. The analysis of these images is entirely within the domain of the specialist, who, based on his/her experience, interprets them and discloses the results to the patient. This paper presents the architecture of a framework that seeks to support the decision-making of medical personnel regarding the diagnosis of diseases. The framework integrates custom tags in the metadata of Digital Imaging and Communications in Medicine(DICOM) files. The tags contain the classification results of supervised learning models. Different convolutional neural network (CNN) architectures trained on medical images were developed using transfer learning and existing pre-trained CNNs to evaluate the framework's performance. A web viewer was also developed to show medical personnel the custom tags. Due to the characteristics of the framework, its use could be extended to patients so that they could obtain a preliminary diagnosis and go to the doctor as soon as possible, which could be crucial.

**Keywords.** DICOM, deep learning, convolutional neural networks, ML.NET, lung cancer.

## 1 Introduction

Cancer is a disease that is becoming more prevalent and is one of the main causes of death worldwide.

The GLOBOCAN 2018 database shows that 2018 saw 18.1 new million cases, with 9.6 million deaths. Lung cancer is the most diagnosed cancer. It has a mortality of 22% in men and 13.8% in women [6].

According to GLOBOCAN 2020 database, there were 19.3 million new cases and 10 million deaths in 2020. The mortality of lung cancer in men was 14.3%, and in women, it was 8.4% for the new cases that occurred in 2020 [31]. Radiologists specializing in medical radiology are the leading actors in detecting and diagnosing lung cancer.

Mastering the skills of a radiologist takes many years of practice; they are taught how to interpret images to diagnose and treat deseases by integrating extensive knowledge of clinical concepts [12]. Additionally, technological advances have allowed artificial intelligence techniques to be applied to detect these conditions [24, 16, 13].

In artificial intelligence, machine learning (ML) plays a leading role due to its high capacity for data processing. Machine learning is based on developing and training algorithms that can infer or predict a result based on a dataset.

Deep learning (DL) is a form of ML based on a multistage array of neural networks that learn from analyzing massive amounts of data. DL employs three main types of learning algorithms:

– **Unsupervised learning,** where data are not categorized, and the algorithm finds patterns that allow the data to be organized in some way;

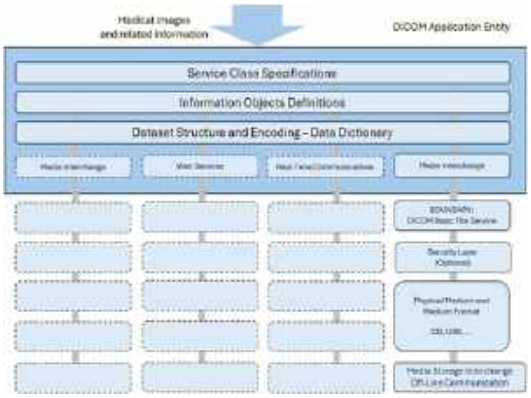



**Fig. 1.** General communication model

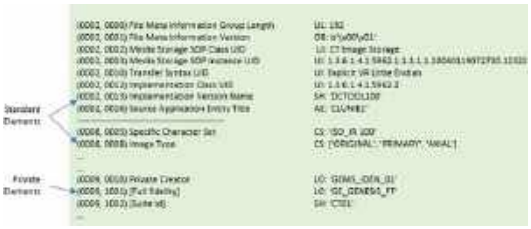



**Fig. 2.** Standard and private DICOM elements

**Fig. 3.** DICOM file showing patient-identifying data (patient's name)

**Fig. 4.** File from Figure 3 with patient-identifying data (patient's name) replaced with "**.**" string

– **Semi-supervised learning,** which uses partially labeled datasets;
– **Supervised learning,** which depends on the labels given to the training data.

DL includes several supervised learning techniques, such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), and Deep Neural Networks (DNNs) [1]. Among the applications of CNNs, their extensive application in diagnosing medical images stands out.

The transfer learning (TL) technique applies a model pre-trained on millions of images from one domain to another domain with a smaller set of images. This technique favors the rapid development of models that provides the same performance results as the model trained with the massive dataset [15].

Some contributions from the scientific community concerning detecting cancer automatically using different classification algorithms are described below. Ramteke and Monali [26] propose an image classification method to classify images into two classes, normal and abnormal, based on the characteristics of the images and the automatic detection of abnormalities. The method consists of four main steps: a) preprocessing, b) feature extraction, c) classification, and d) post-processing. The K-nearest neighbor (KNN) algorithm is employed and is compared with a support vector machine (SVM) based image classifier. KNN achieves an accuracy of 80%, much better than the 69% accuracy obtained by the SVM.

Masood et al. [19] propose a computerized assistance system to support radiologists in lung cancer diagnosis based on DL using a dataset from the Medical Body Area Network (MBAN). This DFCNet model uses a fully convolutional neural network (FCNN), which is utilized to classify each detected spot in four stages of lung cancer. The effectiveness of the proposed work is assessed on different datasets with varying scanning conditions. Overall, the accuracies of CNN and DFCNet were 77.6% and 84.58%, respectively.

The experimental results illustrate the significance of the proposed method for detecting and classifying lung cancer nodules. Miah and

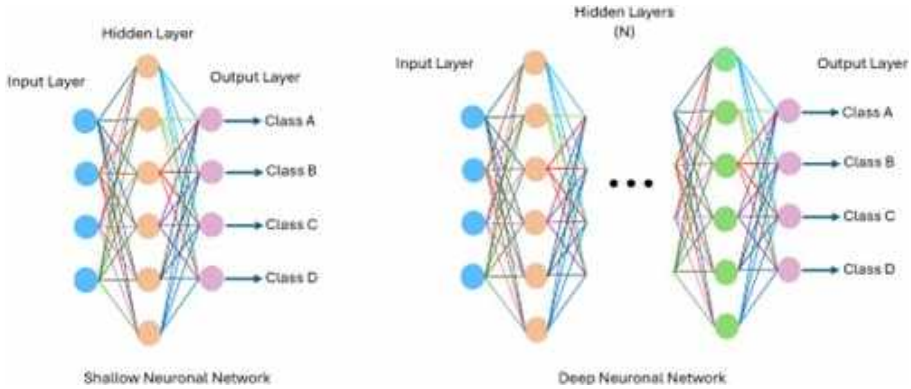



**Fig. 5.** Shallow neural network and deep neural network

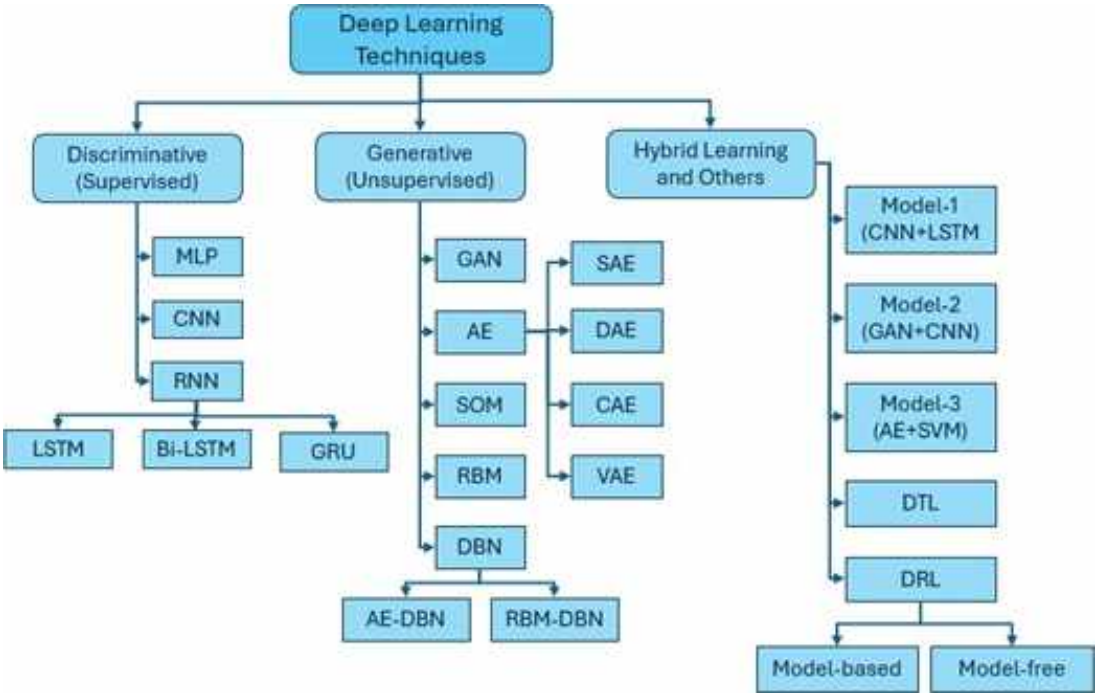



**Fig. 6.** Branches of deep learning

Yousuf [20] present a lung cancer detection model using computed tomography (CT) images and image processing and neural networks. In this approach, the dataset is preprocessed using digital image recognition algorithms, the segmentation of areas of interest, and the classification of these segments using convolutional neural networks.

In the first step, a binary conversion technique detects cancer with a comparison value. In the second step, the image with cancer is segmented, and a feature extraction method is applied. These segments are used to train a neural network, and then, the system is tested with images with and without cancer. This system achieves an accuracy of 96.67%.

Sasikala et al. [28] utilize a CNN to categorize lung tumors as benign or malignant. This approach is based on taking regions of interest from the image; then, every slice is segmented to find tumors. The accuracy obtained with this method is

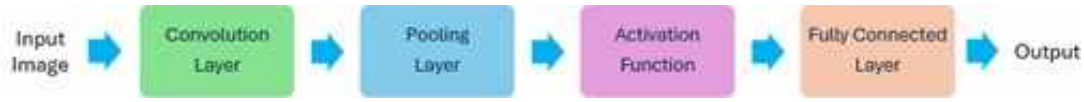



**Fig. 7.** Components of a CNN

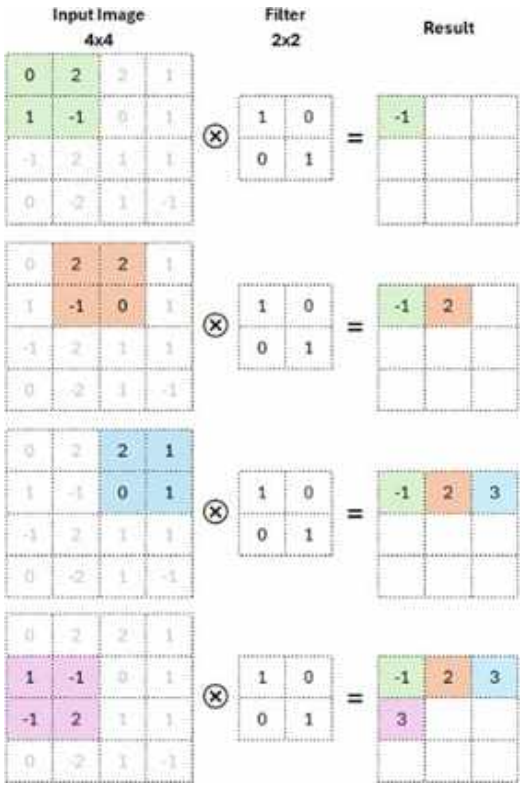



**Fig. 8.** Application of the filter or kernel to the input image

96%; it is more efficient than other traditional neural network methods.

The dataset is obtained from the Lung Image Database Consortium (LIDC) and the Image Database Resource Initiative (IDRI). Shaziya [30] proposes an automatic classification and detection system for lung cancer in medical images using DL. With a CNN model, the proposed method is meant to categorize spots on the lungs in pulmonary CT images from the LIDC dataset.

A total of 6,691 images containing nodules and non-nodules are provided as input to a four-layer 2D CNN model. The model is trained on 70% of the dataset, validated on 10% of the dataset, and tested on 20% of the dataset. The evaluation conducted on the test data resulted in an accuracy of 93.58%, a sensitivity of 95.61%, and a specificity of 90.14%.

On the other hand, the progress of information technology in the medical sector has required the development of communication protocols or standards for managing information in a simple, secure, and comprehensive manner. The most widely used protocol in the medical sector is DICOM. It addresses five general application areas:

1. Online image management.
2. Online image interpretation.
3. Online image printing.

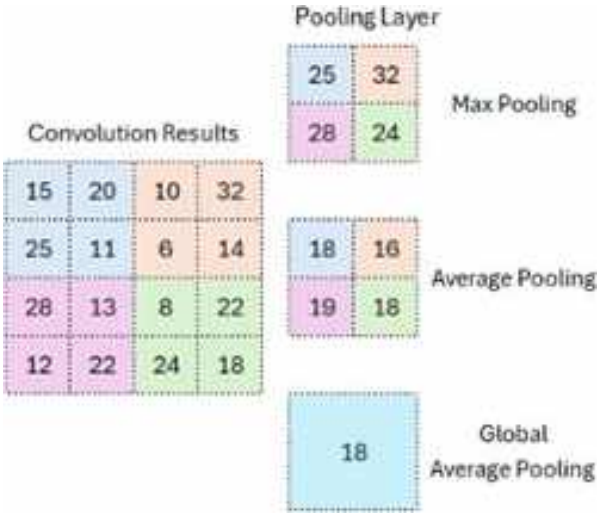



**Fig. 9.** Pooling layer

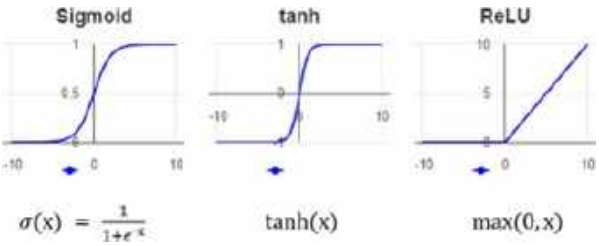



**Fig. 10.** Activation function

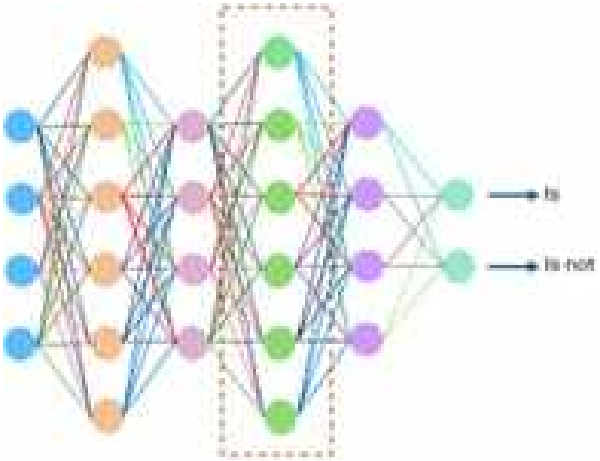


**Fig. 11.** Fully connected layer

4. Online image procedure management.
5. Offline storage media management.

This standard is a comprehensive specification of the elements necessary to achieve a practical level of automatic interoperability among biomedical imaging systems. DICOM provides detailed engineering information that can be used in interface specifications to enable connectivity between various pieces of vendor equipment.

The standard describes how to format and exchange the associated medical image information within and outside the hospital (e.g., teleradiology and telemedicine, among others) [5]. Among the research works that have utilized the DICOM standard is that of Angarita et al. [2], in which the MÉDICO MWEB system is described.

This system was developed with a data structure based on the DICOM standard model, tools (enhancement tools, measurements tools, filters) for visualization and analysis, an intuitive exploration and navigation system for image collection accessible via the web with any browser, and other added features.

A three-layer architecture, a design that introduces an intermediate layer into the process, was used for project development. In this type of architecture, each level is given a simple task, allowing the design of scalable architectures, i.e., they can be easily expanded if the requirements change.

Through the application, DICOM files can be uploaded to the public or private directory of the user, and it also manages an interface for managing the fields of the file; fields can be added, modified, and deleted.

Similarly, DICOM files can be created from JPG images, registering basic standard data that will be attached to the image in the DICOM file. All processes are handled through the JDT library and with an interface developed in JSP and Ajax.

Archie and Marcus [3] describe the DICOM browser application as a software system that views and modifies DICOM file information. Its installation requires the user to have computer knowledge beyond primary computer usage. This application is part of the XNAT software system, defined as an open-source application available for generic use in medical applications.

Similarly, installing XNAT requires advanced computer usage knowledge. XNAT presents

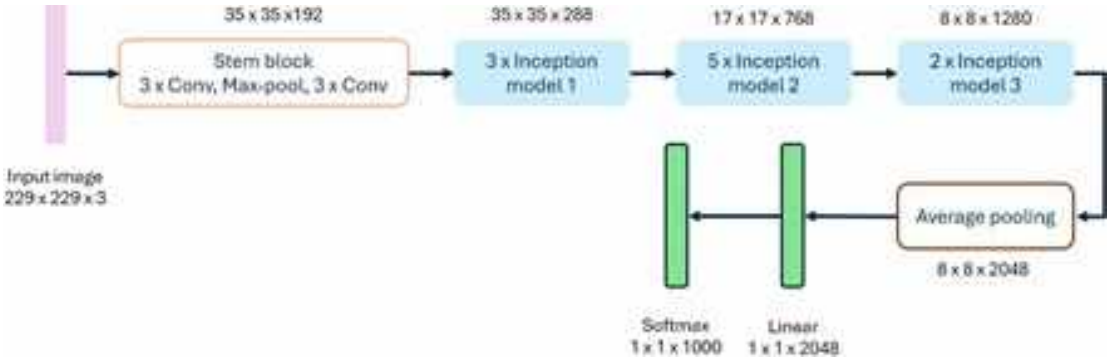



**Fig. 12.** Structure of InceptionV3 model

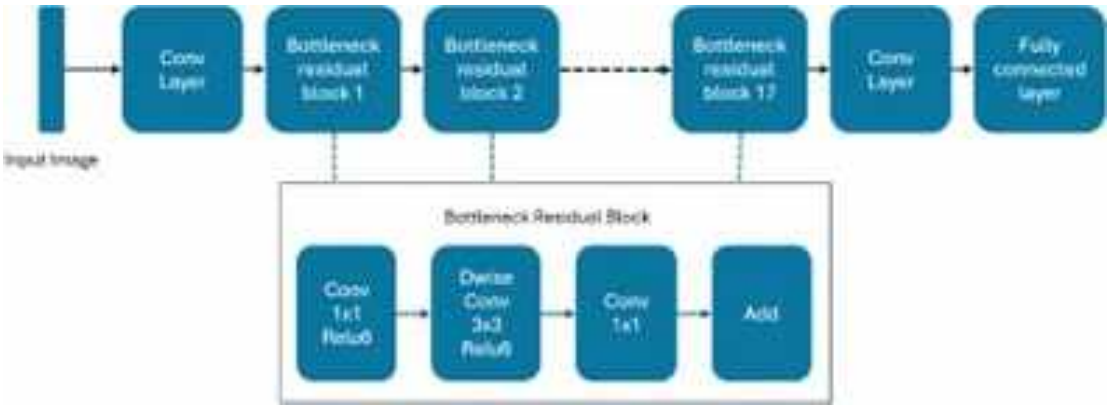



**Fig. 13.** Structure of MobileNetV2

a series of steps for installing a pre-trained deep learning model. Doing this requires the user to learn hardware techniques (NVIDIA) and advanced configurations.

Castro et al. [8] present a DICOM image viewer based on a hybrid architecture that uses client-server, model-view-view-model (MVVM), and N-layer architectural patterns. The client-server style defines a relationship between two applications in which one sends requests to another for processing.

The fundamental concept of MVVM is to separate the model from the view by introducing an abstract layer that allows more accessible and more scalable management of interaction and states. For the development of the client-server application, the HTML5 and JavaScript libraries were used on the client side, and C# with .NET Framework version 4 was used on the server side. Other JavaScript libraries that were used include WADO and KnockoutJS.

Vellez et al. [33] describe Visilab Viewer as a web application that adheres to the DICOM standard. It uses a Flask REST API architecture, Waitress as a WSGI server, and PyTorch as a library for deduction using DL techniques due to its widespread use in both research and commercial applications and because of the ability to import models from other systems.

For CNNs to make deductions, it is necessary to obtain image segments that fit an image with a specific magnification and divide them into patches of the size requested by the CNN. Finally, inference will be achieved by applying a diagnostic rule. Vellez et al. developed a server that manages the Difference in Proportions of Labels (DPL) module using Python 3 and Flask, as it

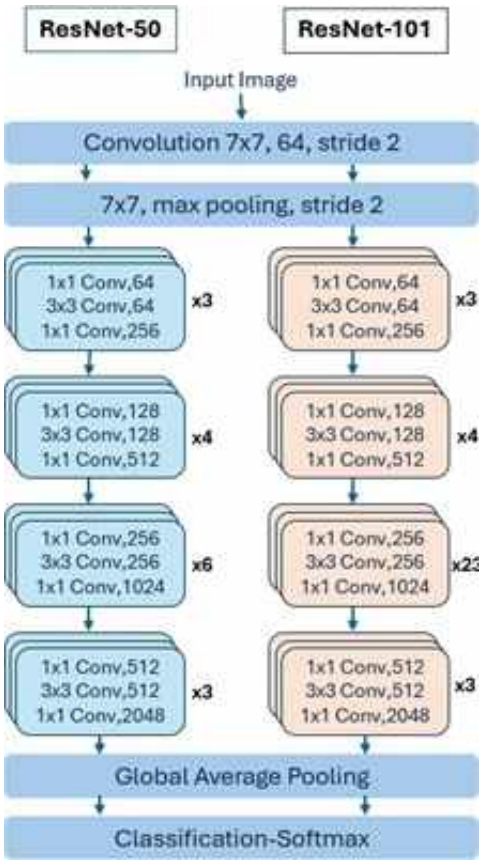



**Fig. 14.** Layers of ResNet50 and ResNet101

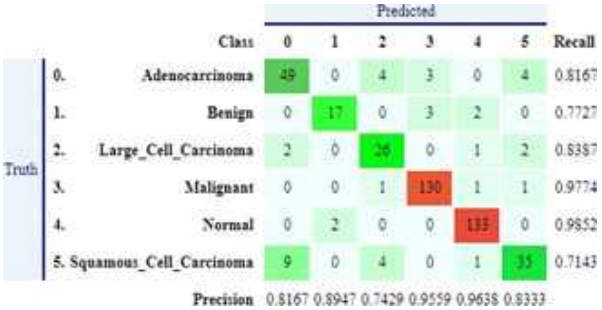



**Fig. 15.** Confusion matrix

natively allows multiple requests to be responded to simultaneously.

Thus, the system can have numerous users simultaneously or receive different inquiries from the same user. This system uses a database with breast cancer images and three different models, which are HER2 classification, the Ki67

**Table 1.** CT scan images of lung cancer dataset class names and indexes

| Name | Index | Assigned Name |
|---|---|---|
| Adenocarcinoma_left.lower. lobe_T2_N0_M0_Ib | 0 | Adenocarcinoma |
| Benign | 1 | Benign |
| Large.cell.carcinoma_left. hilum_T2_N2_M0_IIIa | 2 | Large_Cell_Carcinoma |
| Malignant | 3 | Malignant |
| Normal | 4 | Normal |
| Squamous.Cell.squamous. cell.carcinoma_left. hilum_T1_N2_M0_IIIa | 5 | Squamous_Cell_Carcinoma |

**Table 2.** IQ-OTHNCCD Lung Cancer dataset class names

| Name | Index | Assigned Name |
|---|---|---|
| Benign | 0 | Benign |
| Malignant | 1 | Malignant |
| Normal | 2 | Normal |

proliferation index, and tumor area detection in H&E WSI using the following neural networks: AlexNet (AN), GoogLeNet (GN), VGG-16 (VGG), ResNet-101 (RN), and DenseNet-201 (DN).

Pham et al. [23] present the VinDr system, which has two branches related to the classification of CT images of the chest VinDr-ChestCT and XR images of the chest VinDr-ChestXR. This system focuses on identifying various parts of the body; it is a DL classifier that takes an unknown X-ray as an input image and classifies it into one of five groups, including abdominal X-rays, adult chest X-rays, pediatric chest X-rays, spine X-rays, and others.

From a functional standpoint, a reliable DICOM image router must ensure two essential requirements, including (1) an approximately 100% classification accuracy and (2) providing fast inference.

Mathematically, this supervised multiclass classification task assigns a class label to each input sample. In the present work, a method is proposed to assist the radiologist in

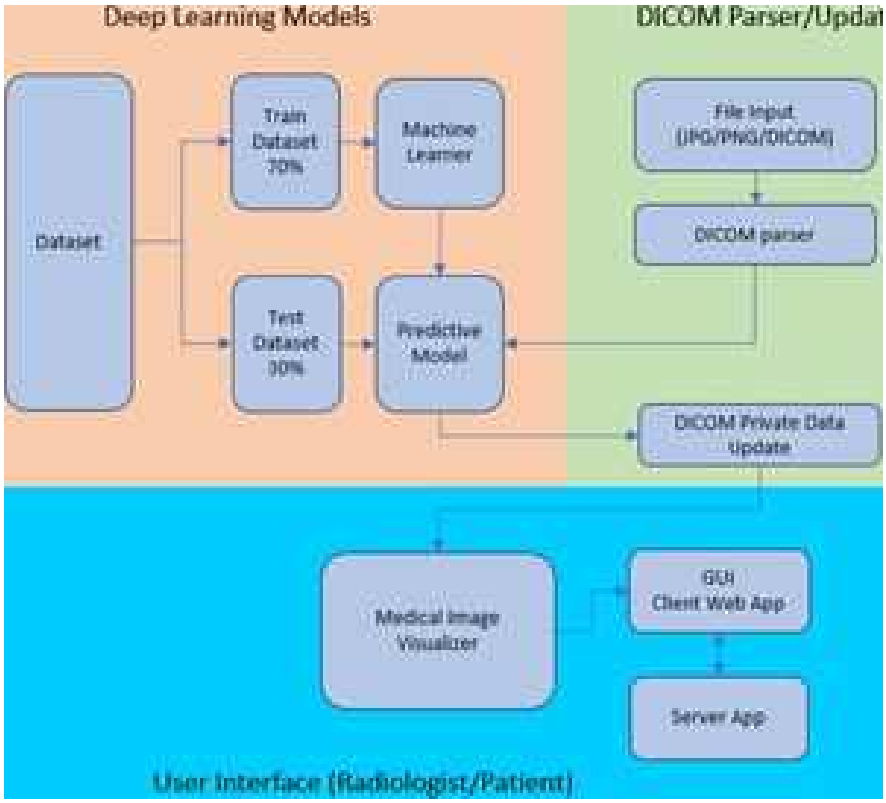



**Fig. 16.** Proposed architecture

decision-making concerning the diagnosis of lung medical images.

This method consists of an architecture that integrates a) deep learning models, b) custom private DICOM tags, and c) a viewer for displaying classification results. This paper is organized as follows. Section 2 presents the basic theory of this research. Section 3 describes the proposed architecture. Section 4 presents the experiments conducted. Lastly, Section 5 presents the conclusions and future work.

## 2 Background

In this section, the related concepts for this research are presented. Section 2.1 describes the DICOM standard. Section 2.2 introduces the concept of "anonymization" which relates to security and confidentiality for the patient, the radiologist, and all personnel involved in the review and classification of medical images.

Section 2.3 describes the architecture of the convolutional neural networks used in deep learning. Section 2.4 describes transfer learning. Section 2.5 discusses the Machine Learning .NET library (ML.NET). Section 2.6 discusses deep learning models in ML.NET. Lastly, Section 2.7 discusses evaluation metrics for machine learning models.

### 2.1 The DICOM Standard

DICOM[1] is a crucial concept in the world of digital imaging. The absence of a standard inhibits usability and the exchange of images, forcing users to deal with many data formats and convert data from one format to another.

[1] www.dicomstandard.org

**Table 3.** DICOM private element x0055, 0010

| Private Tag | Description | Data |
|---|---|---|
| x0055,0010 | Private Creator | UACJ_VISOR |
| x0055,1010 | Model | InceptionV3.zip |
| x0055,1011 | Dataset | IQ-OTHNCCD Lung Cancer Dataset |
| x0055,1012 | Date | 2024 04 21 09:23:52.123 |
| x0055,1013 | FileName | 000160.png |
| x0055,1014 | FileSize | 89.364kB |
| x0055,1015 | Class | Prediction(%) |
| x0055,1016 | Malignant | 99.01 |
| x0055,1017 | Benign | 0.99 |
| x0055,1018 | Normal | 0 |
| x0055,1019 | Model | MobilenetV2.zip |
| x0055,101a | Dataset | IQ-OTHNCCD Lung Cancer Dataset |
| x0055,101b | Date | 2024 04 21 09:23:52.665 |
| x0055,101c | FileName | 000160.png |
| x0055,101d | FileSize | 89.364kB |
| x0055,101e | Class | Prediction(%) |
| x0055,101f | Malignant | 100 |
| x0055,1020 | Benign | 0 |
| x0055,1021 | Normal | 0 |
| x0055,1022 | Model | ResnetV2101.zip |
| x0055,1023 | Dataset | IQ-OTHNCCD Lung Cancer Dataset |
| x0055,1024 | Date | 2024 04 21 09:23:56.053 |
| x0055,1025 | FileName | 000160.png |
| x0055,1026 | FileSize | 89.364kB |
| x0055,1027 | Class | Prediction(%) |
| x0055,1028 | Malignant | 99.98 |
| x0055,1029 | Benign | 0.02 |
| x0055,102a | Normal | 0 |
| x0055,102b | Model | ResnetV250.zip |
| x0055,102c | Dataset | IQ-OTHNCCD Lung Cancer Dataset |
| x0055,102d | Date | 2024 04 21 09:23:57.912 |
| x0055,102e | FileName | 000160.png |
| x0055,102f | FileSize | 89.364kB |
| x0055,1030 | Class | Prediction(%) |
| x0055,1031 | Malignant | 99.99 |
| x0055,1032 | Normal | 0 |
| x0055,1033 | Benign | 0 |

Any image file, in addition to pixel data, contains metadata. Metadata describes the image and plays a significant role in digital imaging. While in general-purpose image formats, metadata may be limited to describing the pixel array, in formats for medical applications, they can describe the image, instrument configuration, image acquisition parameters, and any other elements of interest related to the imaging workflow. The standard helps define the metadata section for the correct use and interpretation of the image.

In the early 1980s, an association of users and healthcare professionals, the American College of Radiology (ACR), and the National Electrical Manufacturers Association (NEMA) began defining a new standard for encoding and exchanging digital medical images. In 1993, the ACR-NEMA committee presented DICOM as a standard with more functionality and long-term vision than previous standardization attempts [18].

Since then, DICOM has been strengthened by including and collaborating with other standards, such as the European Committee for Standardization (CEN) and ISO TC 215 Health Informatics. Figure 1 presents the general communication model for the storage of medical information on any removable media [21].

Applications can use any of the following transport mechanisms: The DICOM message and upper-layer service provides independence from specific physical network support and communication protocols such as TCP/IP.

The DICOM web service API and HTTP service allow the use of common hypertext and the associated protocols for transporting DICOM services. The basic DICOM file service provides access to storage media regardless of specific media storage formats and file structures.

Real-time DICOM communication provides the real-time transport of SMPTE and RTP-based DICOM metadata. The current version of the DICOM standard is composed of the following 22 parts[2]:

– **PS3.1** Introduction and overview.
– **PS3.2** Conformance.

[2]www.dicomstandard.org/current


**Fig. 17.** DICOM x0055, 0010 tag in the panel of the viewer

**Fig. 18.** Aliza MS with 0055—0010 predicted results

- **PS3.3** Information object definitions.
- **PS3.4** Service class specifications.
- **PS3.5** Data structures and encoding.
- **PS3.6** Data dictionary.
- **PS3.7** Message exchange.
- **PS3.8** Network communication support for message exchange.
- **PS3.9** Retired.
- **PS3.10** Media storage and file format for media interchange.
- **PS3.11** Media storage application profiles.
- **PS3.12** Formats and physical media.
- **PS3.13** Retired.
- **PS3.14** Grayscale standard display function.
- **PS3.15** Security and system management profiles.
- **PS3.16** Content mapping resource.
- **PS3.17** Explanatory information.
- **PS3.18** Web services.
- **PS3.19** Application hosting.
- **PS3.20** Imaging reports using HL7 clinical document architecture.
- **PS3.21** Transformations between DICOM and other representations.
- **PS3.22** Real-time communication (DICOM-RTV).

This research focuses on Part 5, Data Structures and Encoding, for accessing standard and private data elements. A data element tag uniquely identifies a data element. Data elements in a dataset shall be ordered by increasing the data element tag number and shall appear at most once in a dataset.

Two types of data elements are defined: 1) standard data elements have an even group number that is not 0000, 0002, 0004, or 0006, and 2) private data elements have an odd group number that is not 0001, 0003, 0005, 0007, or FFFF. The DICOM standard allows the use of standard and private elements as long as they are not already in use.

The reserved elements, both standard and private, are those mentioned above. Figure

2 depicts the structure of two standard data elements, with the group field having even values of 0002 and 0008, and a standard private data element, with the group field having an odd value of 0009.

### 2.2 DICOM Anonymization

DICOM emphasizes the security and protection of the information of the radiologist, patient, and all equipment related to the review and classification of medical images. DICOM establishes in part PS3.15[3] the elements and actions to be executed when anonymizing pertinent information—Table E.1-1a. De-identification Action Codes state the actions on these elements in Table E.1-1.

Application-Level Confidentiality Profile Attributes define the elements and attributes for this purpose. Our proposal adheres to this directive, and to do this, it automatically executes this process when accessing any file with this format. Anonymization consists of removing or replacing all tags specified in Table E.1-1.

Our proposal does not request or store the personal information of any patient, radiologist, doctor, or anyone related to this type of medical activity. If, for any reason, the provided DICOM file contains any of the tags listed in this table, the value of each of these is replaced with a string of the form "**.**"; this indicates that there was a previous value that was replaced by this string.

This string is used only for demonstrative purposes. The application fulfills the directions in Table E.1-1a. None of these tags are removed from the original file. If this framework application updates any of these tags, a new file is generated by adding the following name ending:

"-Anonymous", between the original name and its extension type. Figure 3 displays an original DICOM file with some patient information in it. Figure 4 shows the result from this framework, which shows that confidential patient information was replaced with the dummy string "**.**".

[3]dicom.nema.org/medical/dicom/current/output/chtml/part15/chapter_E.html\#table_E.1-1

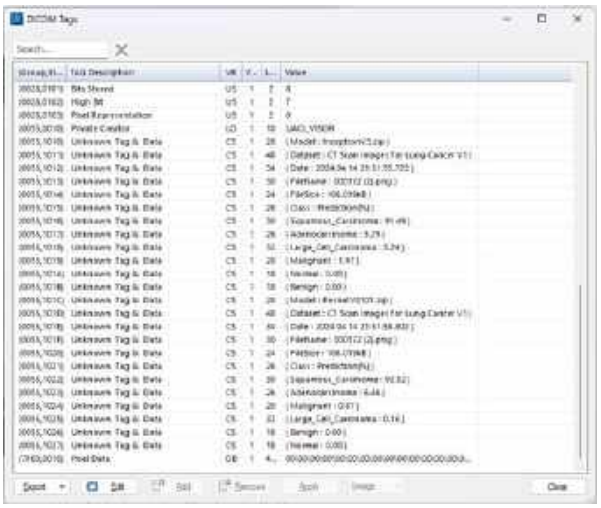


**Fig. 19.** MicroDICOM panel with (0055,0010) classification results


**Fig. 20.** Selection of the image classification task through model builder

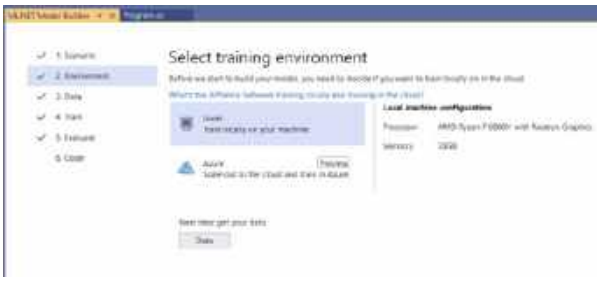



**Fig. 21.** Selection of the development environment

ISSN 2007-9737




**Fig. 22.** Data selection


**Fig. 23.** Model selection and training

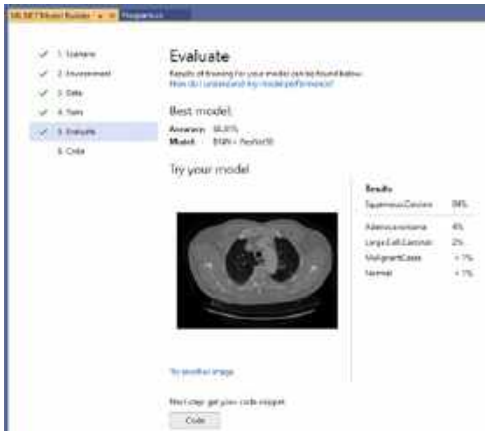



**Fig. 24.** Evaluation of the trained model and test image

**Fig. 25.** Options for reusing the trained model

### 2.3 Deep Learning

Deep learning, a branch of ML and AI, is regarded today as a core component of the current Fourth Industrial Revolution (4IR or Industry 4.0). DL technology originated from artificial neural networks (ANN), and due to its ability to process and learn from data, it has become a significant topic in computer science; it has been widely applied in various areas such as healthcare, visual recognition, text analysis, and cybersecurity. However, building a reasonable DL model is difficult due to the constantly changing nature and variations in real-world problems and data. Sarker et al. [27] illustrate the difference between a shallow neural network (SNN) and a DNN, where an SNN has only one layer. The DNN consists of multiple layers, as shown in Figure 5. Similarly, Sarker et al. [27] define the following categories of DL:

1. Supervised: Uses labeled training data.
2. Unsupervised: Utilize unlabeled datasets.
3. Semi-supervised: Combines both supervised and unsupervised.
4. Reinforcement: Approach focused on the context of the considered problem.

Deep learning is divided into the following three branches:

1. DNN with supervised/discriminative learning.
2. DNN with unsupervised/generative learning.
3. Hybrid learning combining the above models, as shown in Figure 6.

CNNs are based on multi-layer neural networks that can identify, recognize, and classify objects and detect and segment objects in images. The CNN is a well-known architecture of discriminative DL that can learn straight from the input object without requiring human involvement for feature extraction. Figure 7 shows the basic structure of a CNN [32]. A convolutional neural network consists of a convolutional layer, poling, an activation function, and a fully connected layer.


**Fig. 26.** Open-source software used

**Fig. 27.** Interaction between the client and the server

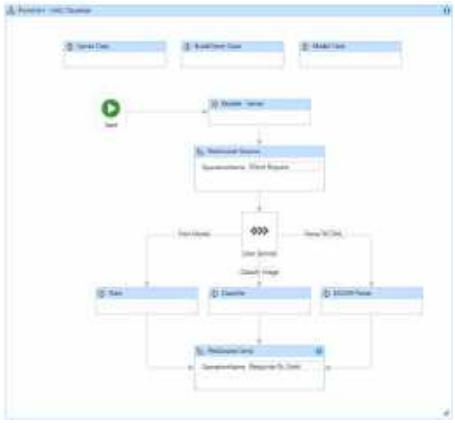


**Fig. 28.** Server-side classes

**Fig. 29.** DICOM viewer GUI

– **Convolutional layer:** This step applies filters to the input data (input image). The kernel is a set of integer values. The CNN filter weights are a set of randomly chosen integers. The kernel learns to extract significant features because these weights are modified during training. It calculates the inner product of the images of all data pairs in the feature space. This mechanism is shown in Figure 8.
– **Pooling:** This is used to reduce the size of the feature map once the filter has been applied. Down-sampling is an essential part of pooling, which helps decrease the upper layers' complexity. The number of filters is not affected by it. The max pooling method is one of the most used methods.

  The image is divided into rectangular subregions, and the maximum value within each subregion is selected. A standard max pooling size is $2 \times 2$. As shown in Figure 9, when pooling is applied in the upper-left corner, the operation shifts to the upper-right corner and moves by two steps. The filter moves in $2 \times 2$ steps to perform pooling.
– **Activation function:** The non-linearity layer allows the generated output to be changed. This layer is used to limit or oversaturate the output. Each activation function in a neural network fulfills the essential process of mapping the input to the output.

  The input value is calculated as the weighted sum of the neuron's input and its bias. This bias implies that the activation function decides whether a neuron is activated in response to a given input, generating the corresponding output. Figure 10 shows the most common activation functions.
– **Fully connected layer:** This step arranges neurons in groups. As shown in Figure 11, every node in every layer is connected directly to another node in the previous layer and next layer.

### 2.4 Transfer Learning

DL has two types of TL: feature extraction and fine-tuning. A dataset like ImageNet is used for feature extraction, but the top layer used for

**Table 4.** InceptionV3 CT scan images of lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Adenocarcinoma | 147 | 48 | 195 |
| Benign | 55 | 25 | 80 |
| Large_Cell_Carcinoma | 81 | 34 | 115 |
| Malignant | 316 | 144 | 460 |
| Normal | 316 | 139 | 455 |
| Squamous_Cell_Carcinoma | 115 | 40 | 155 |
| Total (Images) | 1030 | 430 | 1460 |
| Size (MB) | 134.36 | 59.07 | 193.43 |

**Table 5.** InceptionV3 IQ-OTHNCCD lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Benign | 90 | 30 | 120 |
| Malignant | 393 | 168 | 561 |
| Normal | 293 | 123 | 416 |
| Total (Images) | 776 | 321 | 1097 |
| Size (MB) | 111.12 | 46.52 | 157.63 |

**Table 6.** InceptionV3 CT scan images of lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.9942 |
| microAccuracy | 0.9163 |
| macroAccuracy | 0.8471 |
| LogLoss | 0.3053 |
| LogLossReduction | 0.8047 |

classification purposes will be removed. In addition to the pre-trained model, a new classifier is trained to complete the classification task.

The pre-trained model is considered an arbitrary feature extractor that extracts valuable features from the new dataset. For fine-tuning, the weights of the pre-trained model are taken as the initial values for the latest training and are reworked and fine-tuned in the process.

In this case, the weights are adjusted from generic feature maps to specific attributes related to the new dataset. Fine-tuning aims to adapt the generic features to a particular task instead of overriding generic learning [29]. The work [25] describes how Resnet50V2 was trained using the ImageNet dataset. Databases from different sources are used to retrain existing models like the ones mentioned here. One of these sources is Kaggle, where there are datasets with CT medical images of various types of cancer. Additionally, there are open-source libraries such as Microsoft's Machine Learning .NET (ML.Net) library, which provides support for applying transfer learning from an application developer's standpoint.

### 2.5 Machine Learning .NET Library

ML.NET is a cross-platform library tool designed to build and train ML models within .NET applications. ML.NET aims to provide the same capabilities data scientists and developers can find in the Python ecosystem. ML.NET is based on the classic ML operation concept: gather data, configure the algorithm, train, and deploy.

ML.NET allows the use of deep learning models such as TensorFlow and Open Neural Network Exchange (ONNX), enabling developers to train CNN classification models. The entire ML.NET library is built on the .NET Core framework [11].

### 2.6 Deep Learning Models with ML.NET

Below are descriptions of the deep learning models InceptionV3, MobileNetV2, ResNetV2101, and ResNetV250:

– **InceptionV3:** The InceptionV3 model [7] utilizes convolutional filters of different sizes, allowing it to obtain receptive fields of other areas. To reduce the design space of the network, it embraces a modular system followed by a final union, thus completing the fusion of features from different scales. This model considers typical congestion and performance problems; better results can be obtained using asymmetric kernels and bottlenecks and by replacing large filters with smaller ones [9]. The configuration of the InceptionV3 model is shown in Figure 12.

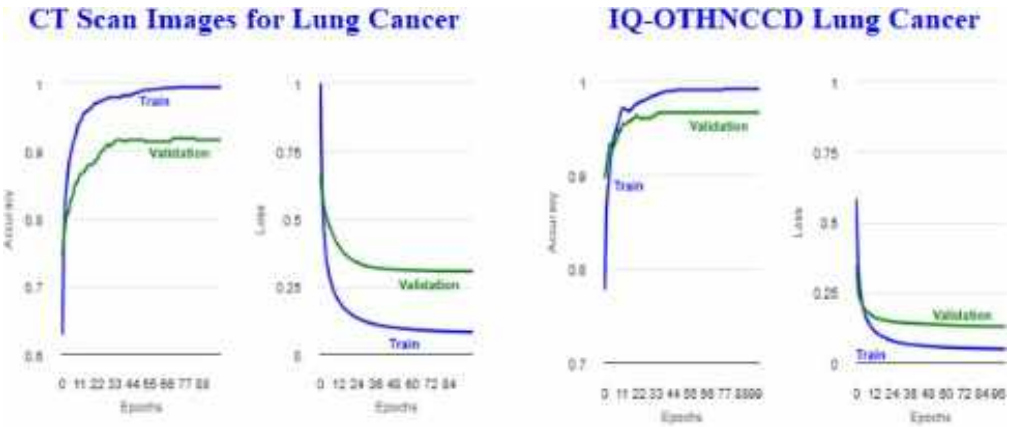



**Fig. 30.** InceptionV3 accuracy and loss plots

– **MobileNetV2:** This model targets portable devices. It is distinct from other CNN architectures, where its links are between bottleneck layers. The middle layer expansion also employs deep levels to filter out non-linear attributes. The MobileNetV2 platform includes 32 convolution layers followed by 19 bottleneck layers. For small datasets, it is not easy to train, and the image classification task becomes challenging.

  This model mitigates this effect by preventing overfitting, and it is a fast and successful architecture that optimizes memory consumption with a low error margin. Additionally, the design of MobileNetV2 provides fast transaction execution during experimentation and optimization of parameters [17]. This model is depicted in Figure 13.

– **ResNetV2101 and ResNetV250:** The Microsoft research team developed ResNet to ease the difficulty of training deeper neural networks. The main idea of ResNet is to learn the additive residual function using shortcut equivalence mappings. It has versions with 18, 34, 50, 101, and 152 weight layers. Instead of learning non-discriminative functions, it utilizes residual functions by adopting skip connections. Unlike VGG, ResNet uses shortcut connections in feedforward neural networks. Figure 14 depicts the layers of these models [4].

### 2.7 Evaluation Metrics for ML Models

Below, we describe the concepts and metrics used to assess the performance of machine learning models. Most metrics use relevant information from the confusion matrix about the algorithm and classification rules. This matrix registers the differences between the actual (rows) and predicted (columns) classifications [14] , as shown in Figure 15. The following metrics are calculated using values from the confusion.

**Precision:** It is the fraction of true positive (TP) parts divided by the total number of units predicted positively (column sum of predicted positives).

True positives are the parts that have been labeled as positive by the model and are positive. False positives (FP) are the parts labeled as positive by the model that are actually negative [14]:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{1}$$

**Recall:** This is the fraction of true positives divided by the total number of positive elements (sum of rows of true positives). Specifically, false negatives (FN) are the elements labeled as false by the model that are actually positive [14]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{2}$$

**Accuracy:** The sum of true positives (TP) and true negatives (TN) in the numerator is

**Table 7.** InceptionV3 IQ-OTHNCCD lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.9923 |
| microAccuracy | 0.9657 |
| macroAccuracy | 0.9373 |
| LogLoss | 0.1309 |
| LogLossReduction | 0.8589 |

**Table 8.** InceptionV3 CT scan images of lung lancer confusion matrix

| Class-Truth | 0 | 1 | 2 | 3 | 4 | 5 | Recall | LogLoss |
|---|---|---|---|---|---|---|---|---|
| 0 | 37 | 0 | 3 | 2 | 1 | 5 | 0.7708 | 0.8114 |
| 1 | 0 | 19 | 0 | 2 | 4 | 0 | 0.76 | 0.6629 |
| 2 | 3 | 0 | 28 | 0 | 0 | 3 | 0.8235 | 0.905 |
| 3 | 0 | 0 | 0 | 144 | 0 | 0 | 1 | 0.0363 |
| 4 | 0 | 3 | 0 | 0 | 136 | 0 | 0.9784 | 0.0997 |
| 5 | 7 | 0 | 2 | 0 | 1 | 30 | 0.75 | 0.6467 |
| Precision | 0.7872 | 0.8636 | 0.8485 | 0.973 | 0.9577 | 0.7895 | | |

**Table 9.** InceptionV3 CT scan images of lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Adenocarcinoma | 0.7872 | 0.7708 | 0.7789 | 48 |
| Benign | 0.8636 | 0.76 | 0.8085 | 25 |
| Large_Cell_Carcinoma | 0.8485 | 0.8235 | 0.8358 | 34 |
| Malignant | 0.973 | 1 | 0.9863 | 144 |
| Normal | 0.9577 | 0.9784 | 0.968 | 139 |
| Squamous_Cell_Carcinoma | 0.7895 | 0.75 | 0.7692 | 40 |
| Accuracy | | | 0.9163 | 430 |
| Macro avg | 0.8699 | 0.8471 | 0.8578 | 430 |
| Weighted avg | 0.914 | 0.9163 | 0.9148 | 430 |

**Table 10.** InceptionV3 IQ-OTHNCCD lung cancer confusion matrix

| Truth | Class | 0 | 1 | 2 | Recall | LogLoss |
|---|---|---|---|---|---|---|
| 0 | Benign | 26 | 0 | 4 | 0.8667 | 0.6534 |
| 1 | Malignant | 0 | 167 | 1 | 0.994 | 0.0163 |
| 2 | Normal | 6 | 0 | 117 | 0.9512 | 0.1599 |
| | precision | 0.8125 | 1 | 0.959 | | |

**Table 11.** InceptionV3 IQ-OTHNCCD lung cancer classification report

| Column1 | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Benign | 0.8125 | 0.8667 | 0.8387 | 30 |
| Malignant | 1 | 0.994 | 0.997 | 168 |
| Normal | 0.959 | 0.9512 | 0.9551 | 123 |
| Accuracy | | | 0.9657 | 321 |
| Macro avg | 0.9238 | 0.9373 | 0.9303 | 321 |
| Weighted avg | 0.9668 | 0.9657 | 0.9662 | 321 |

**Table 12.** MobileNetV2 CT scan images of lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Adenocarcinoma | 143 | 52 | 195 |
| Benign | 52 | 28 | 80 |
| Large_Cell_Carcinoma | 86 | 29 | 115 |
| Malignant | 324 | 136 | 460 |
| Normal | 313 | 142 | 455 |
| Squamous_Cell_Carcinoma | 112 | 43 | 155 |
| Total (Images) | 1030 | 430 | 1460 |
| Size (MB) | 133.47 | 59.96 | 193.43 |

**Table 13.** MobileNetV2 IQ-OTHNCCD lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Benign | 86 | 34 | 120 |
| Malignant | 404 | 157 | 561 |
| Normal | 286 | 130 | 416 |
| Total (Images) | 776 | 321 | 1097 |
| Size (MB) | 111.05 | 46.59 | 157.63 |

**Table 14.** MobileNetV2 CT scan images of lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.997 |
| microAccuracy | 0.9116 |
| macroAccuracy | 0.8422 |
| LogLoss | 0.2656 |
| LogLossReduction | 0.8314 |

divided by all entries in the confusion matrix. TP and TN, found on the main diagonal, represent correctly classified instances. Accuracy reflects the probability that the model's prediction is correct [14]:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \quad (3)$$

**F1-score (binary case):** It is the weighted average between precision and recall, where the best value of the F1-score is one and its worst value is zero. The contribution of precision and recall are the same int the F1-score, and the harmonic mean helps find the best proportion between the two quantities [14]. The F1-score will detect any weaknesses in the prediction algorithm if any such weaknesses exist:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (4)$$

**F1-score (multiclass case):** For multiclass cases, the F1-score involves all classes. To achieve this, we need a multiclass measure of precision and recall to be inserted into the harmonic mean. These metrics can have two distinct specifications, resulting in two other metrics: the micro F1-score and macro F1-score.

For the calculation of the macro and micro F1-score, the precision and recall are now needed for all classes. Formulas (5) and (6) illustrate the calculation of precision and recall for a generic class k [14]:

$$\text{Precision}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k}, \quad (5)$$

$$\text{Recall}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FN}_k}. \quad (6)$$

**Macro F1-score:** The macro average precision and macro average recall are needed to calculate this parameter. Formulas (7) and (8) describe these metrics; they are calculated as the arithmetic mean of the metrics for individual classes.

Formula (9) presents the macro F1-score function. Macro Average precision (MAP), Macro Average recall (MAR) and MacroF1-score (MF1-score) are defined as:

$$\text{MAP} = \frac{\sum_{k=1}^{K} \text{precision}_k}{k}, \quad (7)$$

$$\text{MAR} = \frac{\sum_{k=1}^{K} \text{recall}_k}{k}, \quad (8)$$

$$\text{MF1} - \text{score} = 2 \times \frac{\text{MAP} \times \text{MAR}}{\text{MAP} + \text{MAR}}. \quad (9)$$

**Micro F1-score:** To obtain the micro F1-score, micro-average precision, and micro-average recall should be calculated first. It considers all units together without regard to possible class differences.

These metrics are calculated as follows: It is observed that equations (10) and (11) have the same values; therefore, the average F1 precision is calculated in the same way [14]. Micro Average precision(uAP) and Micro Average recall are defined as:

$$\text{uAP} = \frac{\sum_{k=1}^{K} \text{TP}_k}{\sum_{k=1}^{K} \text{TotalColumn}_k} = \frac{\sum_{k=1}^{K} \text{TP}_k}{\text{GrandTotal}}, \quad (10)$$

$$\text{uAR} = \frac{\sum_{k=1}^{K} \text{TP}_k}{\sum_{k=1}^{K} \text{TotalRow}_k} = \frac{\sum_{k=1}^{K} \text{TP}_k}{\text{GrandTotal}}, \quad (11)$$

$$\text{MicroAverageF1} = \frac{\sum_{k=1}^{K} TP_k}{\text{GrandTotal}}. \quad (12)$$

**LogLoss:** This represents the average logarithmic loss of the classifier. It measures the performance of a classifier based on how much the predicted probabilities diverge from the true class label. A lower value indicates a better model. A perfect model, which predicts a probability of

**Table 15.** MobileNetV2 IQ-OTHNCCD lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.999 |
| microAccuracy | 0.9595 |
| macroAccuracy | 0.887 |
| LogLoss | 0.1247 |
| LogLossReduction | 0.8693 |

**Table 16.** MobileNetV2 CT scan images of lung cancer confusion matrix

| Class-Truth | 0 | 1 | 2 | 3 | 4 | 5 | Recall | LogLoss |
|---|---|---|---|---|---|---|---|---|
| 0 | 45 | 1 | 3 | 0 | 0 | 3 | 0.865 | 0.416 |
| 1 | 0 | 18 | 0 | 0 | 10 | 0 | 0.643 | 0.776 |
| 2 | 3 | 0 | 22 | 0 | 0 | 4 | 0.759 | 0.5 |
| 3 | 0 | 1 | 0 | 131 | 4 | 0 | 0.963 | 0.157 |
| 4 | 0 | 2 | 0 | 0 | 140 | 0 | 0.986 | 0.071 |
| 5 | 6 | 0 | 1 | 0 | 0 | 36 | 0.837 | 0.578 |
| precision | 0.833 | 0.818 | 0.846 | 1 | 0.909 | 0.837 | | |

**Table 17.** MobileNetV2 CT scan images of lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Adenocarcinoma | 0.8333 | 0.8654 | 0.8491 | 52 |
| Benign | 0.8182 | 0.6429 | 0.72 | 28 |
| Large_Cell_Carcinoma | 0.8462 | 0.7586 | 0.8 | 29 |
| Malignant | 1 | 0.9632 | 0.9813 | 136 |
| Normal | 0.9091 | 0.9859 | 0.9459 | 142 |
| Squamous_Cell_Carcinoma | 0.8372 | 0.8372 | 0.8372 | 43 |

**Table 18.** MobileNetV2 IQ-OTHNCCD lung cancer confusion matrix

| Class | Column2 | 0 | 1 | 2 | Recall | LogLoss |
|---|---|---|---|---|---|---|
| 0 | Benign | 23 | 0 | 11 | 0.676 | 0.824 |
| 1 | Malignant | 0 | 157 | 0 | 1 | 0.002 |
| 2 | Normal | 2 | 0 | 128 | 0.985 | 0.089 |
| | precision | 0.92 | 1 | 0.921 | | |

one for the true class, will have a logarithmic loss of zero.

**Macro-accuracy:** It represents the average macro precision of the model. The precision of each class is calculated, and the macro precision is the average of these precisions (macro-average = macro-F1-score).

**Micro-accuracy:** It represents the average micro precision of the model (micro-average).

**Table 19.** MobileNetV2 IQ-OTHNCCD lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Benign | 0.92 | 0.6765 | 0.7797 | 34 |
| Malignant | 1 | 1 | 1 | 157 |
| Normal | 0.9209 | 0.9846 | 0.9517 | 130 |
| Accuracy | | | 0.9595 | 321 |
| Macro avg | 0.947 | 0.887 | 0.9104 | 321 |
| Weighted avg | 0.9595 | 0.9595 | 0.9571 | 321 |

**Table 20.** ResNetV2101 CT scan images of lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Adenocarcinoma | 138 | 57 | 195 |
| Benign | 57 | 23 | 80 |
| Large_Cell_Carcinoma | 84 | 31 | 115 |
| Malignant | 319 | 141 | 460 |
| Normal | 324 | 131 | 455 |
| Squamous_Cell_Carcinoma | 108 | 47 | 155 |
| Total (Images) | 1030 | 430 | 1460 |
| Size (MB) | 137.8 | 55.63 | 193.43 |

**Table 21.** ResNetV2101 IQ-OTHNCCD lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Benign | 90 | 30 | 120 |
| Malignant | 396 | 165 | 561 |
| Normal | 290 | 126 | 416 |
| Total (Images) | 776 | 321 | 1097 |
| Size (MB) | 111.18 | 46.45 | 157.63 |

**Table 22.** ResNetV2101 CT scan images of lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.993 |
| microAccuracy | 0.907 |
| MacroAccuracy | 0.8242 |
| LogLoss | 0.2478 |
| LogLossReduction | 0.8435 |

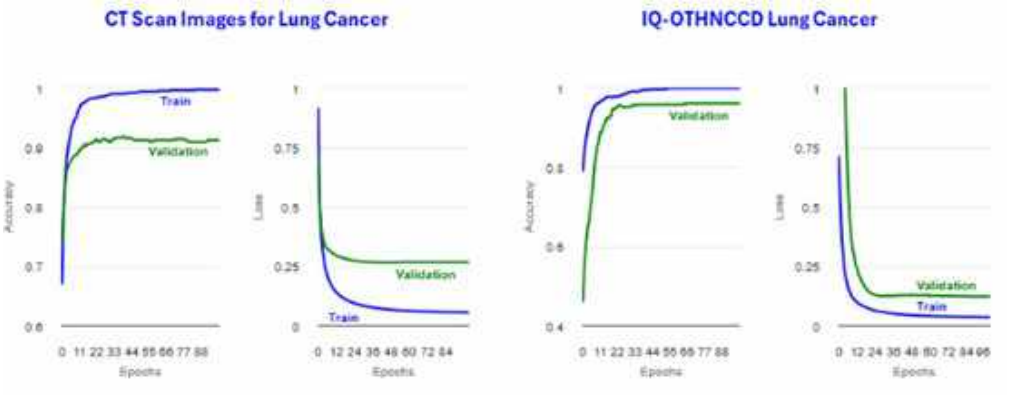



**Fig. 31.** MobileNetV2 accuracy and loss plots

## 3 Proposed Architecture

In this section, the proposed architecture for integrating different CNNs with the custom tags introduced in the DICOM communication standard to support decision-making in lung cancer diagnosis is presented. Two public Kaggle databases containing CT images were used to train the CNN algorithms. The predictions of the algorithms are stored in custom DICOM tags. Section 3.1 presents the proposed architecture. Section 3.2 describes the dataset. Section 3.3 outlines the implementation of DICOM private tags.

Section 3.4 describes the implementation of deep learning models. Lastly, Section 3.5 discusses the interaction between learning models and DICOM.

### 3.1 Description of the Proposed Architecture

The blocks composing our proposal are shown in Figure 16. Three main blocks interact with each other to provide recommendations to the radiologist. The Deep Learning Models block trains the model using the desired algorithm and makes predictions based on the provided image. The DICOM Parser/Updater block facilitates access to the input file to be predicted. It can be a simple image in the JPEG or PNG format or a file containing the entire DICOM standard dataset in addition to the image. This framework does not constrain the file type, size, quality, or consistency.

The User Interface (radiologist/patient) block allows the raiologist or patient to interact with the complete application. It is worth mentioning that the training and prediction tasks are performed on the server where the application runs, while the tasks of displaying results, selecting a DICOM file or image, and executing instructions are performed from the client application. The application code is available at GitHub [4].

### 3.2 Dataset Description

Two different datasets were obtained from the public Kaggle repository: CT [5] Scan Images of Lung Cancer and IQ-OTHNCCD[6] Lung Cancer Dataset. These datasets contain CT medical images of various types of cancer and of healthy individuals. The images are labeled based on the type of disease.

Table 1 shows the original class names of the CT Scan Images of Lung Cancer dataset and the names and index assigned for this research. Table 2 shows the names and index assigned to the IQ-OTHNCCD Lung Cancer dataset that will be used in the remainder of the paper.

[4] github.com/mrodc/uacj-mca-dicom-viewer
[5] www.kaggle.com/datasets/dishantrathi20/ct-scan-images-for-lung-cancer
[6] www.kaggle.com/datasets/hamdallak/the-iqothnccd-lung-cancer-dataset

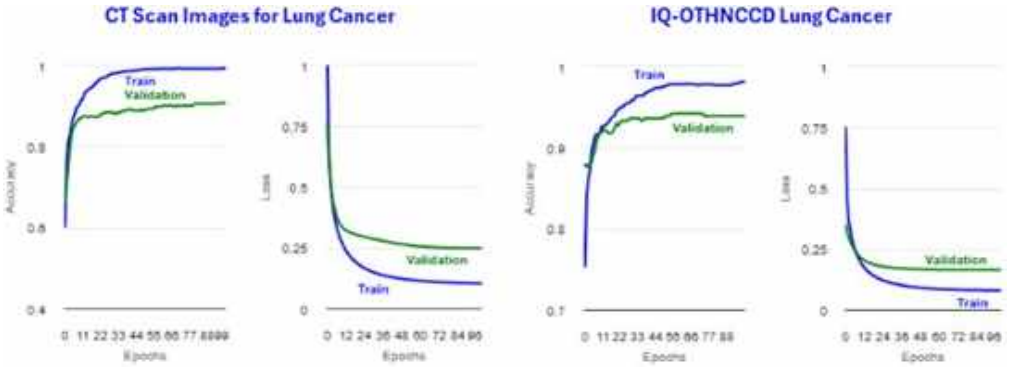



**Fig. 32.** ResNetV2101 accuracy and Loss plots

### 3.3 Implementation of DICOM Private Tags

DICOM private tags are elements that do not have any meaning or encoding in the standard. This self-registration scheme allows each developer to define their own set of private data, the meaning of which must be published in the provider's DICOM documentation.

Developers can document essential values in a structured way in these private elements [10]. The DICOM standard defines private elements and establishes an effective way to use them. These private elements contain information not in standard data elements, such as manufacturer-specific information [22].

In our proposal, the DICOM standard was applied to use these private tags to store the prediction of each pre-trained model in a different private data element. The tag x0055 was defined as a private element for registering the prediction results of the trained models.

It is worth mentioning that the framework verifies whether this tag is in use; if it is, a new one is calculated by incrementing by two until another tag is available. Table 3 shows the element x0055, 0010 added to the DICOM private elements.

These added custom tags are used to show the percentage probability of each class, predicted by each deep-learning model. This information is shown for each model in the implemented viewer.

In Table 3, the tag x0055, 0010 is a private element with the value UACJ_VISOR, which reserves a block of ele-ments. The element x0055, 1010 is part of this block; the "10" in the label of element x0055, 10xx corresponds to the "10" in the label of the private element x0055, 0010.

The predicted results and model are stored starting from private element x0055, 1010. Once the predicted results are stored in these private tags, they are displayed in the viewer's DICOM tag panel, as shown in Figure 17.

Because the predicted results are stored under the DICOM standard, these are also available to other DICOM viewers. Figure 18 shows these values in the Aliza MS application. The Figure 19 shows them in the MicroDicom viewer.

### 3.4 Implementation of Deep Learning Models

In this section, the implementation of CNNs using transfer learning is presented. The implemented models were InceptionV3, MobileNetV2, Res-NetV2101 and ResNetV250, which are available in the ML.NET library. The models were trained with the following parameter settings, Epoch-100, BatchSize-25, LearningRate-0.01, TestFraction-0.3, and TrainFraction-0.7.

This parameter configuration is done in ML.NET code. With these TestFraction and TrainFraction parameters, the subsets for testing and training contain 430 images and 1030 images, respectively. The Microsoft Visual Studio 2022 Community version was used to perform the transfer learning process, which integrates the Model Builder option to leverage pre-trained machine learning models included in ML.NET.

The tasks performed in ML.NET include binary classification, multiclass classification,

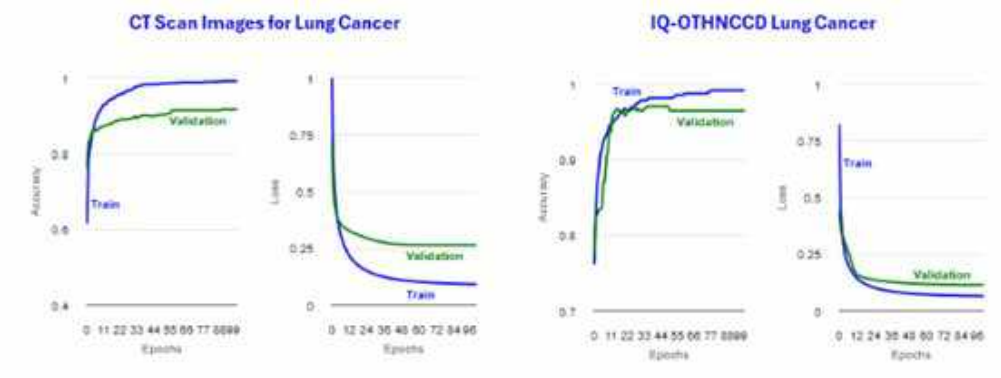



**Fig. 33.** ResNetV250 accuracy and loss plots

image classification, text classification, regression, recommendation, and forecasting. The transfer learning process was performed for the four models and involved the six steps described below:

1. **Selection of the scenario:** The first step in initiating the transfer learning process was to select the task. Figure 20 shows the selection of the image classification task locally through Model Builder.

2. **Selection of the training environment:** In this step, the following options are available: local graphics processing unit (GPU), local central processing nit (CP), and the cloud with Microsoft Azure services, as shown in Figure 21. The local option was selected.

3. **Add dataset:** In this step, we select the path where the database containing the CT medical images with lung cancer is stored, as illustrated in Figure 22.

4. **Train the model:** In this step, the pre-trained model will now be trained with the selected database. The training process takes place once the model is chosen, as shown in Figure 23.

5. **Evaluation model:** Once the model is trained, its evaluation is carried out. Figure 24 presents an example of how the results of a model are shown; this model reached an accuracy of 88.81%. It also shows the percentage probability of having the diseases represented by each class: 94% Squamous.Carcinoma, 4% Adenocarcinoma, 2% Large.Cell.Carcinoma, less than 1% Malignant, and less than 1% Normal. The label is not displayed when the probability is too small, as with Benign.

6. **Code:** Once the model is trained and its performance is known, it can be re-used. For this, Model Builder provides three options: (1) Console App, which allows it to be reused in a console application, (2) Web API, which allows it to be reused in a web application, and (3) the generation of a Notebook. This is shown in Figure 25. The developer determines the option to use. The chosen option was the Console App.

### 3.5 Interaction Between Learning Models and DICOM

The architecture implementation relies on open-source software such as Microsoft's .NET libraries, C# libraries from fo-dicom on the server-side, and JavaScript libraries like jQuery.js and Dojo.js for client-side development, as shown in Figure 26. The server and client interaction is established through a web socket, allowing bidirectional communication.

This means that the server can send notifications to the connected user without waiting for the client to send a communication request. In this structure, the server trains the

**Table 23.** ResNetV2101 IQ-OTHNCCD lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.981 |
| microAccuracy | 0.9377 |
| MacroAccuracy | 0.8631 |
| LogLoss | 0.1675 |
| LogLossReduction | 0.82 |

**Table 24.** ResNetV2101 CT scan images of lung cancer confusion matrix

| Class-Truth | 0 | 1 | 2 | 3 | 4 | 5 | Recall | LogLoss |
|---|---|---|---|---|---|---|---|---|
| 0 | 47 | 0 | 3 | 0 | 0 | 7 | 0.825 | 0.476 |
| 1 | 0 | 14 | 0 | 1 | 8 | 0 | 0.609 | 0.834 |
| 2 | 4 | 0 | 21 | 1 | 0 | 5 | 0.677 | 0.948 |
| 3 | 0 | 0 | 0 | 140 | 1 | 0 | 0.993 | 0.039 |
| 4 | 0 | 4 | 0 | 0 | 127 | 0 | 0.97 | 0.076 |
| 5 | 4 | 0 | 1 | 1 | 0 | 41 | 0.872 | 0.328 |
| precision | 0.854 | 0.778 | 0.84 | 0.979 | 0.934 | 0.774 | | |

**Table 25.** ResNetV2101 CT scan images of lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Adenocarcinoma | 0.8545 | 0.8246 | 0.8393 | 57 |
| Benign | 0.7778 | 0.6087 | 0.6829 | 23 |
| Large_Cell_Carcinoma | 0.84 | 0.6774 | 0.75 | 31 |
| Malignant | 0.979 | 0.9929 | 0.9859 | 141 |
| Normal | 0.9338 | 0.9695 | 0.9513 | 131 |
| Squamous_Cell_Carcinoma | 0.7736 | 0.8723 | 0.82 | 47 |
| Accuracy | | | 0.907 | 430 |
| Macro avg | 0.8598 | 0.8242 | 0.8382 | 430 |
| Weighted avg | 0.9055 | 0.907 | 0.9046 | 430 |

selected TensorFlow model and makes predictions of image pathologies; the client displays the image and the model prediction results to the viewer. The interaction between the server and the client is shown in Figure 27.

The server-side application is composed of three classes: the Model class, BuildDicom class, and Server class, as shown in Figure 28. The

**Table 26.** ResNetV2101 IQ-OTHNCCD lung cancer confusion matrix

| Truth | Class | 0 | 1 | 2 | Recall | LogLoss |
|---|---|---|---|---|---|---|
| 0 | Benign | 20 | 1 | 9 | 0.667 | 0.677 |
| 1 | Malignant | 1 | 164 | 0 | 0.994 | 0.038 |
| 2 | Normal | 8 | 1 | 117 | 0.929 | 0.215 |
| precision | | 0.69 | 0.988 | 0.929 | | |

**Table 27.** ResNetV2101 IQ-OTHNCCD lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Benign | 0.6897 | 0.6667 | 0.678 | 30 |
| Malignant | 0.988 | 0.9939 | 0.9909 | 165 |
| Normal | 0.9286 | 0.9286 | 0.9286 | 126 |
| Accuracy | | | 0.9377 | 321 |
| Macro avg | 0.8687 | 0.8631 | 0.8658 | 321 |
| Weighted avg | 0.9368 | 0.9377 | 0.9372 | 321 |

**Table 28.** ResNetV250 CT scan images of lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Adenocarcinoma | 136 | 59 | 195 |
| Benign | 62 | 18 | 80 |
| Large_Cell_Carcinoma | 90 | 25 | 115 |
| Malignant | 331 | 129 | 460 |
| Normal | 316 | 139 | 455 |
| Squamous_Cell_Carcinoma | 95 | 60 | 155 |
| Total (Images) | 1030 | 430 | 1460 |
| Size (MB) | 136.03 | 57.4 | 193.43 |

**Table 29.** ResNetV250 IQ-OTHNCCD lung cancer image distribution

| Class | Train | Test | Total |
|---|---|---|---|
| Benign | 87 | 33 | 120 |
| Malignant | 392 | 169 | 561 |
| Normal | 297 | 119 | 416 |
| Total (Images) | 776 | 321 | 1097 |
| Size (MB) | 112.3 | 45.34 | 157.63 |

**Table 30.** ResNetV250 CT scan images of lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.991 |
| microAccuracy | 0.9163 |
| macroAccuracy | 0.8633 |
| LogLoss | 0.262 |
| LogLossReduction | 0.8333 |

**Table 31.** ResNetV250 IQ-OTHNCCD lung cancer metrics

| Metric | Value |
|---|---|
| Accuracy | 0.991 |
| microAccuracy | 0.9626 |
| macroAccuracy | 0.8934 |
| LogLoss | 0.1156 |
| LogLossReduction | 0.877 |

**Table 32.** ResNetV250 CT scan images of lung cancer confusion matrix

| Class-Truth | 0 | 1 | 2 | 3 | 4 | 5 | Recall | LogLoss |
|---|---|---|---|---|---|---|---|---|
| 0 | 46 | 0 | 1 | 1 | 1 | 10 | 0.78 | 0.654 |
| 1 | 0 | 14 | 0 | 1 | 3 | 0 | 0.778 | 0.839 |
| 2 | 2 | 0 | 20 | 0 | 0 | 3 | 0.8 | 0.545 |
| 3 | 0 | 0 | 0 | 127 | 2 | 0 | 0.984 | 0.048 |
| 4 | 0 | 3 | 0 | 1 | 135 | 0 | 0.971 | 0.072 |
| 5 | 6 | 0 | 2 | 0 | 0 | 52 | 0.867 | 0.487 |
| precision | 0.852 | 0.824 | 0.87 | 0.977 | 0.957 | 0.8 | | |

Model class contains the methods and logic for training the desired model.

The BuildDicom class contains the methods for interpreting DICOM tags. This class implements editing both standard tags and private tags. It also incorporates the ability to process more than one image per file, known as multi-frame imaging.

The Server class describes the multiprocessing method that serves each user connected through

**Table 33.** ResNetV250 CT scan images of lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Adenocarcinoma | 0.8519 | 0.7797 | 0.8142 | 59 |
| Benign | 0.8235 | 0.7778 | 0.8 | 18 |
| Large_Cell_Carcinoma | 0.8696 | 0.8 | 0.8333 | 25 |
| Malignant | 0.9769 | 0.9845 | 0.9807 | 129 |
| Normal | 0.9574 | 0.9712 | 0.9643 | 139 |
| Squamous_Cell_Carcinoma | 0.8 | 0.8667 | 0.832 | 60 |
| Accuracy | | | 0.9163 | 430 |
| Macro avg | 0.8799 | 0.8633 | 0.8707 | 430 |
| Weighted avg | 0.9161 | 0.9163 | 0.9157 | 430 |

**Table 34.** ResNetV250 IQ-OTHNCCD lung cancer confusion matrix

| Truth | Class | 0 | 1 | 2 | Recall | LogLoss |
|---|---|---|---|---|---|---|
| 0 | Benign | 23 | 2 | 8 | 0.697 | 0.63 |
| 1 | Malignant | 0 | 169 | 0 | 1 | 0.025 |
| 2 | Normal | 2 | 0 | 117 | 0.983 | 0.102 |
| | Precision | 0.92 | 0.988 | 0.936 | | |

**Table 35.** ResNetV250 IQ-OTHNCCD lung cancer classification report

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Benign | 0.92 | 0.697 | 0.7931 | 33 |
| Malignant | 0.9883 | 1 | 0.9941 | 169 |
| Normal | 0.936 | 0.9832 | 0.959 | 119 |
| Accuracy | | | 0.9626 | 321 |
| Macro avg | 0.9481 | 0.8934 | 0.9154 | 321 |
| Weighted avg | 0.9619 | 0.9626 | 0.9604 | 321 |

**Table 36.** Metrics of deep learning methods on CT scan images for lung cancer dataset

| Metric | InceptionV3 | MobileNetV2 | ResNetV2101 | ResNetV250 |
|---|---|---|---|---|
| Accuracy | 0.994 | 0.997 | 0.993 | 0.991 |
| microAccuracy | 0.9163 | 0.9116 | 0.907 | 0.9163 |
| MacroAccuracy | 0.8471 | 0.8422 | 0.8242 | 0.8633 |
| LogLoss | 0.3053 | 0.2656 | 0.2478 | 0.262 |
| LogLossReduction | 0.8047 | 0.8314 | 0.8435 | 0.8333 |

**Table 37.** Metrics of deep learning methods on IQ-OTHNCCD lung cancer dataset

| Metric | InceptionV3 | MobileNetV2 | ResNetV2101 | ResNetV250 |
|---|---|---|---|---|
| Accuracy | 0.992 | 0.999 | 0.981 | 0.991 |
| microAccuracy | 0.9657 | 0.9595 | 0.9377 | 0.9626 |
| MacroAccuracy | 0.9373 | 0.887 | 0.8631 | 0.8934 |
| LogLoss | 0.1309 | 0.1247 | 0.1675 | 0.1156 |
| LogLossReduction | 0.8589 | 0.8693 | 0.82 | 0.877 |

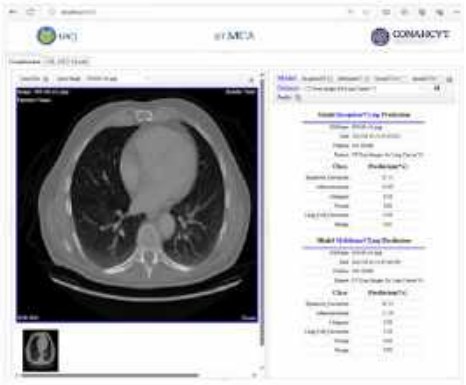


**Fig. 34.** Prediction results of pre-trained InceptionV3 and MobileNetV2 models

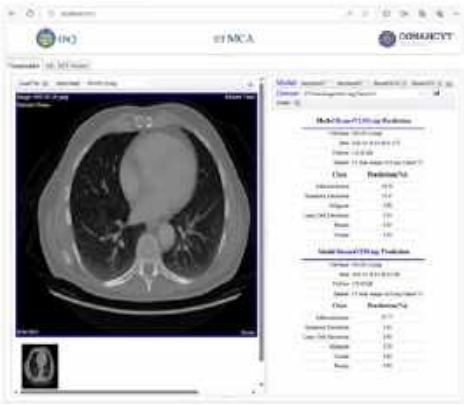


**Fig. 35.** Prediction results of pre-trained ResNetV2101 and ResNetV250 models

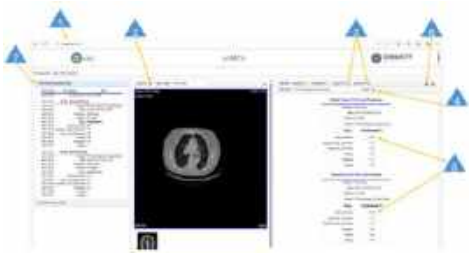


**Fig. 36.** DICOM viewer user interface

a web socket. The user interface of the developed viewer shows the private elements (0055, xxxx) described in Table 3. Figure 29 shows the user interface of the viewer.

## 4 Results

This section presents the results obtained from the proposed architecture. Section 4.1 describes the results obtained by each of the models. Section 4.2 shows the structure of the DICOM viewer. Lastly, Section 4.3 describes the steps to classify a medical image or DICOM file.

### 4.1 Results Obtained

The performance achieved by each of the trained models is presented below. The reported metrics are AccuracyMacro, AccuracyMicro, Recall, Precision, LogLoss, and the Confusion Matrix and the Classification Report are also provided.

For all trained models, the LogLoss value is better the closer it is to zero. For Precision and Recall, a value closer to one is better. We ran each model ten times per dataset.

The results with higher microAccuracy values are presented below. It is worth mentioning that the tables and figures of the results presented in this section were generated by the viewer developed in this work. Tables 4 and 5 present InceptionV3 image distribution files. Tables 6 and 7 display the metrics for the training accuracy (Accuracy) and test accuracy (microAccuracy).

Figure 30 shows the plots obtained for the accuracy and loss. Tables 8 and 10 display confusion matrices. Tables 9 and 11 show the classification reports. Tables 12 and 13 present MobileNetV2 image distribution files. Tables 14 and 15 display the metrics for the trainig accuracy (Accuracy) and test accuracy (microAccuracy).

Figure 31 shows the plots obtained for the accuracy and loss. Tables 16 and 18 display confusion matrices. Tables 17 and 19 show the classification reports. Tables 20 and 21 present ResNetV2101 image distribution files.

Tables 22 and 23 display the metrics for the training accuracy (Accuracy) and test accuracy (microAccuracy). Figure 32 shows the plots

obtained for the accuracy and loss. Tables 24 and 26 display confusion matrices. Tables 25 and 27 show the classification reports.

Tables 28 and 29 present ResNetV250 image distribution files. Tables 30 and 31 display the metrics for the training accuracy (Accuracy) and test accuracy (microAccuracy). Figure 33 shows the plots obtained for the accuracy and loss. Tables 32 and 34 display confusion matrices.

Tables 33 and 35 show the classification reports. Tables 36 and 37 summarize the model metrics. We observe that for CT Scan Images for the Lung Cancer dataset, the positive prediction percentage is above 90.0%.

The ResNetV2101 model obtained the lowest value of 90.7%, while the ResNetV250 model achieved the highest value of 91.63%, InceptionV3 obtained 91.63%; and MobileNetV2 obtained 91.16%. Even though InceptionV3 and ResNetV250 have the same value, the difference in the accuracy value is bigger for InceptionV3. Thus, we can conclude that these models have a good prediction performance.

For the IQ-OTHNCCD Lung Cancer dataset, the positive prediction percentage is above 93.70%. The InceptionV3 model obtained the highest value of 96.57%, and ResNetV2101 had the lowest value of 93.77%. MobileNet ob-tained 95.95%, while ResNetV250 obtained 96.26%.

### 4.2 Medical Image DICOM Viewer

After the training and evaluation phase, the predictions of the models are displayed using the DICOM standard. Figure 34 shows the prediction results of the pre-trained InceptionV3 and MobileNetV2 models. Figure 35 shows the prediction results of the pre-trained ResNetV2101 and ResNetV250 models.

### 4.3 DICOM Viewer User Interface

The DICOM viewer user interface is easy to use. It does not require users to have deep knowledge of deep learning, data science, or computer science. Figure 36 shows the DICOM viewer user interface. The basic steps to perform a prediction task are given below:

1. Type in the web address of the viewer.
2. Click the Load File button to load an image file (JPG/PNG format) or a DICOM file (.dcm extension).
3. Select the model or models for the prediction task. This is achieved by clicking on the check box on the model task bar.
4. Click the Predict button to run the prediction process. This will take a few seconds to finish.
5. This step involves reading the reported values. The closer they are to 100%, the higher the probability that the pathology is present for this image.
6. Click on the disk icon to save the obtained results. This creates or updates a DICOM-format file, which including the private element x0055, 0010.
7. The DICOM file tags are available on the DICOM Standard Data panel.

## 5 Conclusion

In this research, we present the architecture of a decision support method to assist radiologists in diagnosing pathologies in medical images, focusing on detecting lung cancer with six different pathology classes. Our proposed architecture integrates a) deep learning models, b) custom private DICOM tags, and c) a viewer that displays classification results stored in DICOM private tags. The DL models InceptionV3, MobileNetV2, ResNet2101, and ResNetV250 were trained using TensorFlow ML algorithms supported in Microsoft's ML.NET library on the CT Scan Images for Lung Cancer and IQ-OTHNCCD Lung Cancer datasets.

The CT Scan Images for Lung Cancer dataset consists of 1,460 images, with 70% (1030 images) used for training and 30% (430 images) for testing. The IQ-OTHNCCD Lung Cancer dataset is composed of 1,097 images, providing 776 images for training and 321 images for testing using a test fraction of 0.3.

The results show that all the models have an excellent prediction performance above 90%. We added the DICOM private element x0055 to store the prediction results of each trained DL model. The prediction results are displayed for the radiologist and patient through a graphical interface. The graphical interface consists of two main code blocks, one for the client-side application implemented in JavaScript and the other for the server-side application implemented in C#. The proposed architecture shows that it can support radiologists as a second opinion.

In future work, we consider extending the architecture to datasets related to other pathologies. Additionally, algorithms for image analysis can be integrated to perform analysis and add annotations to the image, plus evaluate image quality, resolution, blur, and visibility, among other characteristics. An important future step is to get support and feedback from domain experts (radiologist personnel), as this is necessary to validate the generated prediction results, which should match radiologist's expected results.

## References

1. **Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., Farhan, L. (2021).** Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, Vol. 8, No. 1. DOI: 10.1186/s40537-021-00444-8.
2. **Angarita-Sanguino, C. R., Beltrán-Galvis, N. (2016).** Aplicación web para la visualización de imágenes médicas medicomweb. Respuestas, Vol. 13, No. 2, pp. 32–37. DOI: 10.22463/0122820x.540.
3. **Archie, K. A., Marcus, D. S. (2012).** Dicombrowser: Software for viewing and modifying DICOM metadata. Journal of Digital Imaging, Vol. 25, No. 5, pp. 635–645. DOI: 10.1007/s10278-012-9462-x.
4. **Bayram, B., Kilic, B., Ozoglu, F., Erdem, F., Bakırman, T., Sivri, S., Bayrak, O. C., Delen, A. (2020).** A deep learning integrated mobile application for historic landmark recognition: A case study of Istanbul. Mersin Photogrammetry Journal, Vol. 2, No. 2, pp. 38–50.
5. **Bidgood, W. D., Horii, S. C., Prior, F. W., van-Syckle, D. E. (1997).** Understanding and using dicom, the data interchange standard for biomedical imaging. Journal of the American Medical Informatics Association, Vol. 4, No. 3, pp. 199–212. DOI: 10.1136/jamia.1997.0040199.
6. **Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., Jemal, A. (2018).** Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for Clinicians, Vol. 68, No. 6, pp. 394–424. DOI: 10.3322/caac.21492.
7. **Cao, J., Yan, M., Jia, Y., Tian, X., Zhang, Z. (2021).** Application of a modified inception-v3 model in the dynasty-based classification of ancient murals. EURASIP Journal on Advances in Signal Processing, Vol. 2021, No. 1. DOI: 10.1186/s13634-021-00740-8.
8. **Castro-Márquez, C. L., Delgado-García, A. (2014).** Visor de imágenes médicas digitales web web viewer for digital medical images. Revista Cubana de Informática Médica, Vol. 6, No. 1, pp. 57–70.
9. **Chatterjee, S., Hazra, D., Byun, Y. C., Kim, Y. W. (2022).** Enhancement of image classification using transfer learning and GAN-based synthetic data augmentation. Mathematics, Vol. 10, No. 9, pp. 1541. DOI: 10.3390/math10091541.
10. **Clunie, D. A., Flanders, A., Taylor, A., Erickson, B., Bialecki, B., Brundage, D.,**

**Gutman, D., Prior, F., Seibert, J. A., Perry, J., Gichoya, J. W., Kirby, J., Andriole, K., Geneslaw, L., Moore, S., Fitzgerald, T. J., Tellis, W., Xiao, Y., Farahani, K. (2023).** Report of the medical image de-identification (MIDI) task group-best practices and recommendations.

11. **Esposito, D., Esposito, F. (2022).** Programming ML.NET. Pearson Education, Inc.
12. **European Society of Radiology (2019).** What the radiologist should know about artificial intelligence – an ESR white paper. Insights into Imaging, Vol. 10, No. 1. DOI: 10.1186/s13244-019-0738-2.
13. **Gaytán-Campos, I., Morales-Castro, W., Priego-Sánchez, B., Fitz-Rodríguez, E., Guzmán-Cabrera, R. (2022).** Automatic classification of images with skin cancer using artificial intelligence. Computación y Sistemas, Vol. 26, No. 1, pp. 325–336. DOI: 10.13053/CyS-26-1-4176.
14. **Grandini, M., Bagli, E., Visani, G. (2020).** Metrics for multi-class classification: An overview. arXiv. DOI: 10.48550/arXiv.2008.05756.
15. **Hosna, A., Merry, E., Gyalmo, J., Alom, Z., Aung, Z., Azim, M. A. (2022).** Transfer learning: A friendly introduction. Journal of Big Data, Vol. 9, No. 1. DOI: 10.1186/s40537-022-00652-w.
16. **Jakimovski, G., Davcev, D. (2018).** Lung cancer medical image recognition using deep neural networks. 30th International Conference on Digital Information Management, pp. 1–5. DOI: 10.1109/icdim.2018.8847136.
17. **Kaya, Y., Gürsoy, E. (2023).** A MobileNet-based CNN model with a novel fine-tuning mechanism for COVID-19 infection detection. Soft Computing, Vol. 27, No. 9, pp. 5521–5535. DOI: 10.1007/s00500-022-07798-y.
18. **Larobina, M. (2023).** Thirty years of the DICOM standard. Tomography, Vol. 9, No. 5, pp. 1829–1838. DOI: 10.3390/tomography9050145.
19. **Masood, A., Sheng, B., Li, P., Hou, X., Wei, X., Qin, J., Feng, D. (2018).** Computer-assisted decision support system in pulmonary cancer detection and stage classification on CT images. Journal of Biomedical Informatics, Vol. 79, pp. 117–128. DOI: 10.1016/j.jbi.2018.01.005.
20. **Miah, M. B. A., Yousuf, M. A. (2015).** Detection of lung cancer from CT image using image processing and neural network. International Conference on Electrical Engineering and Information Communication Technology, pp. 1–6. DOI: 10.1109/iceeict.2015.7307530.
21. **National Electrical Manufacturers Association (2024).** Dicom ps3-1. www.dicomstandard.org/current/.
22. **National Electrical Manufacturers Association (2024).** Dicom ps3-5. www.dicomstandard.org/current/.
23. **Pham, H. H., Do, D. V., Nguyen, H. Q. (2021).** DICOM imaging router: An open deep learning framework for classification of body parts from DICOM x-ray scans. medRXiv. DOI: 10.1101/2021.08.13.21261945.
24. **Quintanilla-Domínguez, J., Ruiz-Pinales, J., Barrón-Adame, J. M., Guzman-Cabrera, R. (2018).** Microcalcifications detection using image processing. Computación y Sistemas, Vol. 22, No. 1, pp. 291–300. DOI: 10.13053/cys-22-1-2560.
25. **Rahimzadeh, M., Attar, A. (2020).** A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Informatics in Medicine Unlocked, Vol. 19, pp. 100360. DOI: 10.1016/j.imu.2020.100360.
26. **Ramteke, R., Khachane, M. (2012).** Automatic medical image classification and abnormality detection using K-nearest neighbour. International Journal of Advanced

Computer Research, Vol. 2, No. 4, pp. 190–196.

27. **Sarker, I. H. (2021).** Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science, Vol. 2, No. 6. DOI: 10.1007/s42979-021-00815-1.
28. **Sasikala, S., Bharathi, M., Sowmiya, B. R. (2018).** Lung cancer detection and classification using deep CNN. International Journal of Innovative Technology and Exploring Engineering, Vol. 8, No. 2, pp. 259–262.
29. **Shazia, A., Xuan, T. Z., Chuah, J. H., Usman, J., Qian, P., Lai, K. W. (2021).** A comparative study of multiple neural network for detection of COVID-19 on chest x-ray. EURASIP Journal on Advances in Signal Processing, Vol. 2021, No. 1. DOI: 10.1186/s13634-021-00755-1.
30. **Shaziya, H. (2023).** Automatic detection and classification of lung cancer in pulmonary CT images using deep learning. arXiv. DOI: 10.13140/RG.2.2.22245.37605.
31. **Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., Bray, F. (2021).** Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for Clinicians, Vol. 71, No. 3, pp. 209–249. DOI: 10.3322/caac.21660.
32. **Taye, M. M. (2023).** Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. Computation, Vol. 11, No. 3, pp. 52. DOI: 10.3390/computation11030052.
33. **Vallez, N., Espinosa-Aranda, J. L., Pedraza, A., Deniz, O., Bueno, G. (2023).** Deep learning within a DICOM WSI viewer for histopathology. Applied Sciences, Vol. 13, No. 17, pp. 9527. DOI: 10.3390/app13179527.

*Article received on 28/02/2024; accepted on 14/05/2024.*
**Corresponding author is J. Patricia Sánchez-Solís.*

ISSN 2007-9737

# Using Compensatory Fuzzy Logic to Model an Investor's Preference Regarding Portfolio Stock Selection within Markowitz's Mean-Variance Framework

Luis Cisneros[1], Raul Porras[1], Gilberto Rivera[1,*], Rafael A. Espin-Andrade[2], Vicente Garcia[1]

[1] Universidad Autónoma de Ciudad Juárez, División Multidisciplinaria de Ciudad Universitaria, Mexico

[2] Universidad Autónoma de Coahuila, Facultad de Contaduría y Administración, Mexico

{enrique.cisneros, raul.porras, gilberto.rivera, vicente.jimenez}@uacj.mx, espinr@uadec.edu.mx,

**Abstract.** We analyze the use of Compensatory Fuzzy Logic (CFL) applied to an optimization model to reflect an investor's preferences regarding portfolio stock selection. CFL is a framework that allows the construction of fuzzy predicates using fuzzy parametrized linguistic variables. Although the potential of a CFL predicate to model preferences is high, to the best of our knowledge, this is the first use of this strategy to do so. Real data from the Mexican Stock Exchange was employed to create a test instance. Portfolios were obtained using the Particle Swarm Optimization algorithm. By maximising the degree of truth of the predicate representing the investor's preferences, the model is able to reflect investor profiles regarding the return-risk relation of the portfolios. Three artificial investor profiles were defined during the experimentation; the model was able to reflect all of these preferences.

**Keywords.** Swarm particle optimization, preference incorporation, metaheuristic algorithm, prescriptive analytics, fuzzy optimization.

## 1 Introduction

Investments have been essential in improving welfare levels since the last century [5]. The search for tools to help investors is constituted by the formulation of optimization problems.

These approaches aim to provide solutions that offer a good compromise for the investor in a relatively short computational time and allow the incorporation of criteria close to the reality in which the investor lives day by day.

Markowitz Portfolio Stock Selection (PSS) is a bi-objective optimization problem that maximizes the investment return on a given set of possible investments while minimizing investment risk [14].

In this model, investors seek the highest return possible by using the mean return of each investment. Investors also consider the investment risk in terms of the variance. Thus, a portfolio should balance return and risk.

A portfolio $x$ is defined as a set of $n$ possible investments, where:

$$\sum_{i=1}^{n} x_i = 1, \quad x_i \geq 0. \tag{1}$$

The return $R_i$ for investment $i$ is calculated by the average return of the investment during a time period $T$. The risk is calculated by using the covariance matrix $\sigma$ of $x$. the quality of the portfolio is measured by the following bi-objective function:

$$F = \begin{cases} \max f_1 = \sum_{i=1}^{n} x_i \cdot R_i, \\ \min f_2 = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i \cdot x_j \cdot \sigma(x_i, x_j). \end{cases} \quad (2)$$

Many researchers have addressed this problem by obtaining the Pareto front. However, investors often have preferences regarding the portfolio. Some desire the highest possible return, others the lowest possible risk, and some would rather strike a particular balance between return and risk.

We use Fuzzy Logic (FL) to deal with investors' subjective preferences. FL is a multi-valued logic used to deal with uncertainty [17]. FL is a popular tool to model the preferences of investors [1, 3, 4]. One of the variants of FL is Compensatory Fuzzy Logic (CFL) [7]. CFL involves the use of fuzzy logical operators. These operators allow the construction of fuzzy predicates.

Although CFL predicates may be useful to reflect investor preferences, to our knowledge, there have been no studies using them. Our goal is to use a CFL predicate to guide the algorithm to a portfolio that satisfies a given preference.

This paper is organized as follows. In Section 2, we briefly review the scientific literature. In Section 3, we present some preliminaries for CFL. In Section 4, we introduce the proposed model. In Section 5, we conduct a parametric analysis of the proposed model. Lastly, in Section 6, we discuss the conclusions and directions for future research.

## 2 Related Work

This section presents an overview of the progress in this research topic. Amiri et al. [2] developed two models for portfolio selection. First, a mathematical programming model to maximize the minimum of the Sharpe ratios.

Second, a probabilistic programming model based on the necessity theory, which deals with the uncertainty of the parameters and the low quality of the decisions caused by this same uncertainty.

**Table 1.** Studies in portfolio stock selection

| Study | Strategy | Fuzzy Variable | Preferences |
|---|---|---|---|
| Amiri et al. 2019 | Mathematical Programming Model | Trapezium Distribution | |
| Li et al. 2021 | Genetic Algorithm | Uncertain Distribution | |
| Corazza et al. 2021 | PSO-Dynamic | Risk Measure | ✓ |
| Dai et al. 2021 | Genetic Algorithm | Uncertain Distribution | |
| Harris et al. 2022 | Cubic Spline Smoothing | Utility Values | |
| Thakur et al. 2022 | Ant Colony Optimization | Trapezoidal Numbers | |
| Hamdi et al. 2022 | PSO and ICA | | ✓ |
| Fazli et al. 2022 | Deep Reinforcement Learning | LR Numbers | |
| Nozarpour et al. 2023 | PSO | Triangular and Trapezoidal Numbers | |
| This paper | PSO | Sigmoid | ✓ |

Others who have presented innovative portfolio optimization methods are Li et al. [12], who proposed a three-step model to deal with this problem with different risk preferences. In addition, they used entropy to describe the degree of diversification of portfolio selection to obtain a favorable balance between return and risk.

Corazza et al. [3] proposed solving the unconstrained portfolio selection problem using a hybrid metaheuristic based on Particle Swarm Optimization (PSO) with a dynamic penalty approach, and when compared with the most recent (up to that time) proposals, they concluded their proposal needed only 4% of the computational time consumed by their predecessors to find good compromise solutions to the portfolio selection problem.

Dai et al. [4] proposed a genetic algorithm for the problem of multi-period portfolio optimization in an uncertain environment, where uncertain variants describe the return risks.

Considering the restriction of the minimum number of transaction batches in the real world, they formulated the uncertain mean-VaR model. Moreover, this model is in two concrete forms, assuming the values of the risk have zigzag or regular returns.

Harris et al. [10] explored how an investor with behavioral preferences determined by the cumulative prospect theory can make investment decisions in a range of realistic situations by considering 7 rational benchmarks. In addition, they proposed an alternative behavioral objective function and incorporated the investor's short- and long-term memories into the portfolio decisions.

Thakur et al. [16] proposed a novel fuzzy expert system model to evaluate and rank the stocks in the Bombay Stock Exchange. Evidence from the Dempster-Shafer theory is used to develop a fuzzy rule base to decrease the implementation time and the overall cost of the system.

Hamdi et al. [9] formulated the Conditional Value at Risk using Data Envelopment Analysis (DEA), and then this same model was tested with a PSO algorithm and finally with the Imperial Competitive Algorithm (ICA). They concluded that when DEA is used for portfolio selection modeling, better results are obtained, and the PSO algorithm performs better in portfolio optimization.

A new strategy to address the portfolio management problem is proposed by Fazli et al. [8]. They present a reinforcement learning framework which they use to extract the meaning of the stock price history, and using this information, they generate the vector of weights for the portfolio selection. Nozarpour et al. [15] considered different time horizons for the portfolio assets.

Thus, an investment cannot be traded before a specific point in time, and transaction costs were added to make it more realistic. Table 1 presents some of the significant characteristics of the research reviewed in this paper, such as the strategy used to solve the model, the fuzzy variables that stand out in each approach, and whether or not the preferences of the investor are incorporated in the strategy used.

The PSO and GA are among the most widely used; there are also innovative proposals, such as using neural networks. The fuzzy membership function most used in the state-of-the-art literature is the trapezoidal function. As we can see, there are several studies that incorporate the use of fuzzy logic in the PSS. However, to our knowledge, CFL predicates have yet to explored for to modeling investor preferences.

## 3 Introduction to Compensatory Fuzzy Logic

Compensatory Fuzzy Logic (CFL) is a multivalued logic axiomatic approach based on three operators. These operators correspond to conjunction ($c : [0,1]^n \rightarrow [0,1]$), disjunction ($d : [0,1]^n \rightarrow [0,1]$), and negation ($n : [0,1] \rightarrow [0,1]$) [7]. The operators of conjunction and disjunction can be defined as $t$-norm $T$ functions [7] that have the following properties.

- Commutativity: $T(x, y) = T(y, x)$.
- Monotonicity: $T(x, y) \leq T(u, v)$, if $x \leq u$ and $y \leq v$.
- Associativity: $T(x, T(y, z)) = T(T(x, y), z)$.

CFL also provides some useful definitions: the parametrized sigmoid function and the Generalized Continuous Linguistic Variable (GLCV). The parametrized generalized sigmoid function is defined by:

$$S(x, \alpha, \gamma) = \frac{1}{1 + e^{\alpha(x-\gamma)}}. \tag{3}$$

It should be mentioned that the sigmoid function has several properties that make it favorable for optimization. First, the function is asymptotic in 1. Because of this, a higher value of $x$ will always have a higher degree of truth. Whereas a non-asymptotic function, such as the trapezoidal, doesn't recognize increments regarding the value of $x$ within a certain range.

Second, the asymptotic nature of the sigmoid function complies with the notion of Pareto dominance because a non-dominated solution will always have a higher degree of truth than the solutions it dominates (the better the objective value, the higher degree of truth). Using Equation 3, CFL is able to define the parametrized GLCV as follows [6]:

$$\begin{gathered}\mathrm{GLCV}(x, \alpha, \gamma, m) = \\ (S(x, \alpha, \gamma))^m \cdot (1 - S(x, \alpha, \gamma))^{m-1}.\end{gathered} \tag{4}$$

Each GLCV has three parameters that modify its behavior: $\alpha$ is used to express the level of tolerance of variations in $x$. A high value of $\alpha$ is used to indicate that the tolerance is low, whereas

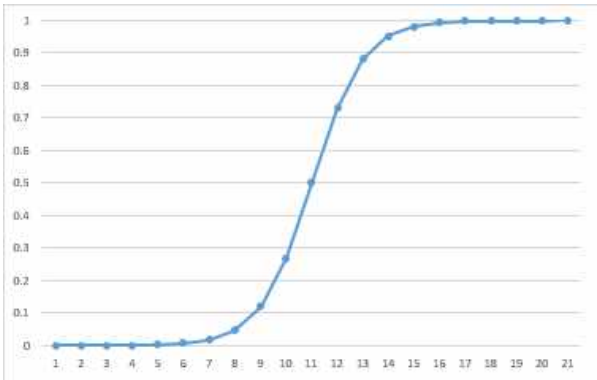


**Fig. 1.** GLCV as a sigmoid function

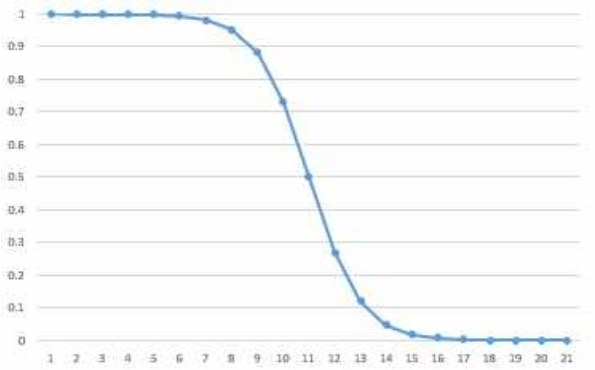


**Fig. 2.** GLCV as an inverse sigmoid function

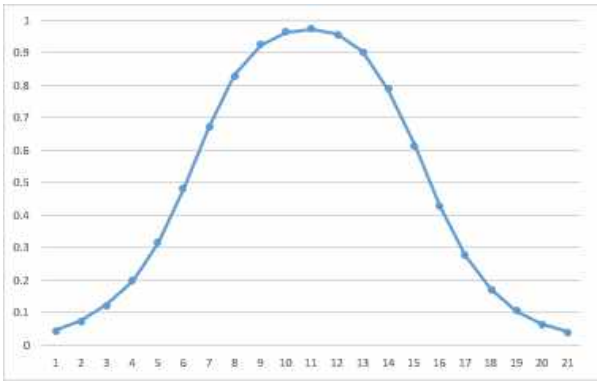


**Fig. 3.** CFL conjunction predicate

a lower value would indicate a higher level of tolerance. $\gamma$ is used to define the magnitude of $x$ at which the truth value of the GLCV is 0.5, while $m$ is used to determine the behavior of the function. With $m = 1$, the GLCV will behave like a sigmoid function, as depicted in Figure 1, which expresses the maximization of the input variable.

On the other hand, $m = 0$ will make the GLCV behave like an inverse sigmoid, as depicted in Figure 2, which expresses minimization. Using GLCVs, we are able to create CFL predicates like the one described in Equation 5:

$$p(x) = \sqrt[n]{\mathrm{GLCV}_1(x) \cdot \mathrm{GLCV}_2(x) \cdots \mathrm{GLCV}_n(x)}. \quad (5)$$

The predicate will behave as a sigmoid function if all of the GLCVs are sigmoids, as an inverse sigmoid if all of the GLCVs are inverse sigmoids, and as a concave function if some GLCVs are sigmoid and others inverse sigmoid. The last case is depicted in Figure 3.

## 4 Proposed Model

To create a test instance, we selected 14 indexes pertaining to the Mexican Stock Exchange. The mean return value for each index was calculated using the daily value at the beginning and the value at the cut from 2021 to 2022.

Then, those indexes with negative returns were discarded. Our proposed model employs the GLCV [6] to create fuzzy predicates that reflect the investor's preference.

### 4.1 CFL Predicate

For the Portfolio Stock Selection (PSS), we defined a predicate $p$ that expresses 'A portfolio that has a high return and a low risk.' Here, we employ two linguistic variables: $A$: 'Portfolio $x$ has a high return,' and $B$: 'Portfolio $x$ has low risk'.

To model these linguistic variables, we use two GLCVs, one for each linguistic variable. Because investors seek to maximize return in all cases, the GLCV assigned to $A$ is set to be a sigmoid, as shown in Equation 6:

$$\mu(A) = \mathrm{GLCV}_i(f_1(x),\, \alpha_i,\, \gamma_i,\, 1). \quad (6)$$

Here, the higher the return, the higher the value of truth the linguistic variable will have. $\gamma_i$ will indicate where the investor starts to consider the return to be 'high'. The other GLCV will model linguistic variable $B$. Unlike the return, the risk is set always to be minimized.

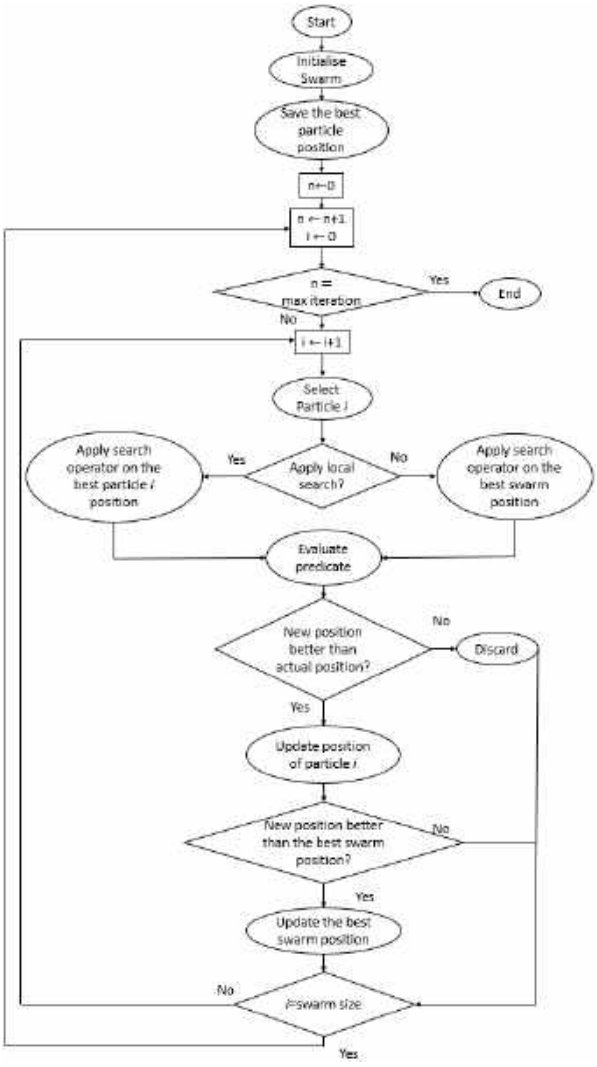



**Fig. 4.** Optimization process

Because of this, we assign an inverse sigmoid to the GLCV, as shown in Equation 7:

$$\mu(B) = \mathrm{GLCV}_j(f_2(x),\, \alpha_j,\, \gamma_j,\, 0). \tag{7}$$

Just as with the previous GLCV, $\gamma_j$ will indicate where the risk starts to be considered low. Then, both GLCVs will be linked using a conjunction operator, as shown in Equation 8:

$$p(x) = \sqrt{\mu(A) \cdot \mu(B)}. \tag{8}$$

Using this predicate, the best portfolio would be one that has a 'high' return and a 'low' risk, with 'high' and 'low' being subjective values.

For example, suppose that an investor has defined the fuzzy parameters as follows:

$$\alpha_i = 1, \qquad \gamma_i = 5, \qquad \alpha_g = 1, \qquad \gamma_i = 2.$$

With this configuration, a portfolio with return $f_1(x) = 4.3$ and risk $f_2(x) = 1.25$ would have the following evaluation:

$$\mu(A) = 0.3318, \quad \mu(B) = 0.6899, \quad p(x) = 0.4711.$$

In this case, the truth value of the predicate is below 0.5, which means that the investor would not be satisfied with this portfolio. However, a portfolio with $f_1(x) = 8.2$ and $f_2(x) = 0.7$ would have the following evaluation:

$$\mu(A) = 0.9608, \quad \mu(B) = 0.7858, \quad p(x) = 0.8689.$$

Here, the increased truth value of the predicate indicates that this portfolio better complies with the investor's preference. By modifying the parameters of the GLCV of Equations 6 and 7, the predicate is able to model the subjective preferences of a wide range of investors.

In the previous example, the investor may change his/her mind about the amount of risk (s)he can tolerate.

Therefore, (s)he could modify $\gamma_j = 2$ to $\gamma_j = 0.5$. With this change, the investor expresses a preference for a portfolio with lower risk. Now the same portfolio with $f_1(x) = 8.2$ and $f_2(x) = 0.7$ is evaluated as follows:

$$\mu(A) = 0.9608, \quad \mu(B) = 0.5498, \quad p(x) = 0.7268.$$

Then, the investor may decide that an acceptable value of the risk should be close to $\gamma_j$. Therefore, (s)he makes the change $\alpha_j = 3$. Now the portfolio is evaluated as follows:

$$\mu(A) = 0.9608, \quad \mu(B) = 0.731, \quad p(x) = 0.838.$$

This predicate will replace Equation 2 as the objective function during the optimization process, becoming the objective function of the PSO.

### 4.2 Particle Swarm Optimization

PSO is a population-based heuristic used to solve optimization problems. The optimization process is depicted in Figure 4. The first step is used to initialize the swarm. This involves creating a new PSS solution for each particle. Within the PSO, we refer to the solution assigned to the particle as its position.

Then, all particles are evaluated by the objective function, the CFL predicate, and the best particle position is saved as the best swarm position. After this step, the optimization process begins.

This process starts by choosing whether the search operator will be applied to the best swarm position or to the best solution found by the particle so far. This selection is done with Equation 9:

$$x_i(t+1) = \begin{cases} s(p,v) & \text{if} R_1 a_p > R_2 a_g, \\ s(g,v) & \text{if} R_1 a_p \leq R_2 a_g, \end{cases} \quad (9)$$

Here, $x_i(t+1)$ denotes the next position of the particle, $R_1$ and $R_2$ are random numbers in the range [0,1], while $a_p$ and $a_g$ are the acceleration constants. Lastly, $v$ is the velocity constant.

The mentioned constants modify the behavior of the swarm. The acceleration ones are used to favor either the selection of the best position of the particle or the best position of the swarm. The velocity constant indicates how many times the search operator is applied.

After creating this new position, the objective function will be used to evaluate it. If the new position is not better than the previous position, it is discarded, and the iteration ends. Whereas if the new position is better, it replaces the current position. Lastly, the position is compared with the best swarm position.

If it is better, it will also replace it, and the iteration ends. This process will continue until PSO reaches the maximum number of iterations previously set by the user.

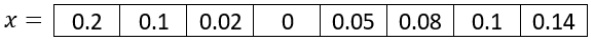



**Fig. 5.** Portfolio example

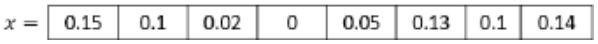



**Fig. 6.** Modified portfolio

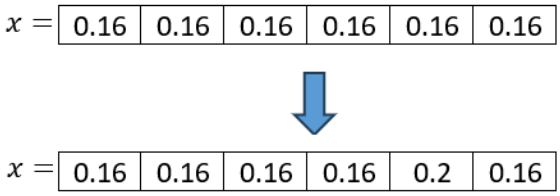



**Fig. 7.** Solution adjustment

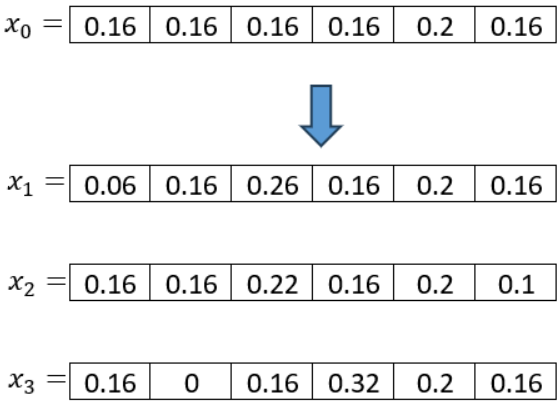



**Fig. 8.** Initial population

#### 4.2.1 Solution Coding

To represent each portfolio, we used a float array where each number represents the percentage of the budget to be invested for index $i$. In Figure 5, we show an example of a solution for the test instance. In Figure 5, we can appreciate how one of the indexes has 0% of the budget. This is an acceptable case within our PSS model, as well as one index having the whole budget.

#### 4.2.2 Search Operator

The search operator creates a new solution from a given solution. To do this, the operator randomly selects an index $x_i$ with a budget percentage over 0%. After selecting the index, a percentage $p$ between 1% and 100% is randomly set to extract from $x_i$. Then, the extracted budget is added to a random index $x_j$.

**Table 2.** Reference portfolios

| | Return | Risk |
|---|---|---|
| High return | 22.5 | -2127.5 |
| Low risk | 13.86 | -4986 |

**Table 3.** Investors percentages

| Profile | Return | | Risk | |
|---|---|---|---|---|
| | $\alpha$ | $\gamma$ | $\alpha$ | $\gamma$ |
| $I_a$ | 99% | 90% | 99% | 70% |
| $I_b$ | 99% | 85% | 99% | 85% |
| $I_c$ | 99% | 70% | 99% | 90% |

**Table 4.** Test parameters

| Parameter | Values | | |
|---|---|---|---|
| Velocity constant | 10 | 5 | 1 |
| Acceleration constant $p$ | 1 | 1.5 | 2 |
| Acceleration constant $g$ | 1 | 1.5 | 2 |

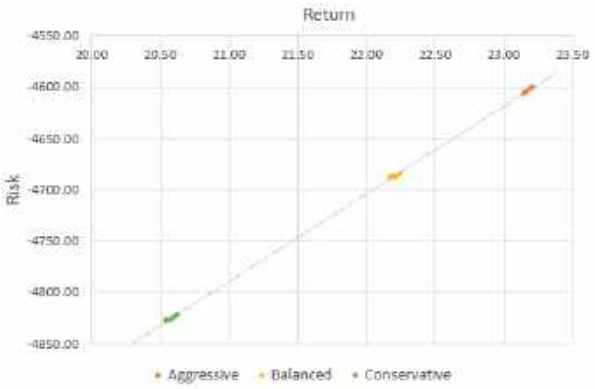



**Fig. 9.** PSO results on combination 17

To illustrate the operator, we will apply it to the portfolio shown in Figure 5. First, $x_1$ is randomly selected. Then, $p$ is randomly set at 25%, corresponding to 0.05 of the budget percentage assigned to $x_1$. After that, $x_5$ is randomly selected to receive the 0.05 from $x_1$.

The budget is assigned to $x_5$ and extracted from $x_1$, resulting in the solution shown in Figure 6. Since the operator bases the budget exchange on percentages, there is no need to satisfy any budget constraint.

#### 4.2.3 Initial Swarm

To create the first swarm, each solution is created by assigning equally the budget to each index. The percentage is calculated by $1/a$, where $a$ is the number of indexes to consider. The result is truncated to two digits. Because of the nature of this operation, it is possible that the total budget could be less than 1.

To correct this, we simply determine the missing budget and randomly assign it to an index. In Figure 7, we show how a solution is created using this method where the number of indexes is 6.

As shown in Figure 7, each index is assigned a budget percentage of 0.16 in the first step. However, when calculating the total budget for this solution, the result is 0.96. Because of this, the remaining 0.04 to reach 1.0 is randomly assigned to the 5th index.

After creating the first solution, the PSO will use it to create the initial population. This is done by applying the local search operator to the initial solution to create each particle. In Figure 8, we show an example of this process.

Because this is the first solution for each particle, it is set as the best personal solution found so far. Each particle is evaluated using the objective function. An implementation of the proposed model is available at GitHub[1].

## 5 Results

To determine the ability of the proposed model to implement multiple preferences, we defined three investor profiles. The first one, $I_a$, is an aggressive investor, meaning that it will prioritize a higher return over the risk. Second, $I_c$ is a conservative investor that prioritizes a low risk. Lastly, $I_b$ is a balanced investor seeking an advantageous trade-off between risk and return. The model was coded in Python 3, and all experiments were run on a Windows 10 PC with 16GB of RAM and a 3.9 GHz CPU. All of the investor profiles will use two portfolios as references. One with high return and one with low risk.

[1] github.com/luis-cis/CFL_Preference_Portfolio_Selection

**Table 5.** Parametric results

| Combination | $I_a$ | $I_b$ | $I_c$ |
|---|---|---|---|
| 1 | 0.5407 | 0.6604 | 0.5513 |
| 2 | 0.9667 | 0.9715 | 0.9244 |
| 3 | 0.9712 | 0.8765 | 0.9062 |
| 4 | 0.8057 | 0.8059 | 0.7817 |
| 5 | 0.9972 | 0.9917 | 0.9937 |
| 6 | 0.9432 | 0.9652 | 0.9583 |
| 7 | 0.9410 | 0.8214 | 0.8615 |
| 8 | 0.9002 | 0.9878 | 0.9597 |
| 9 | 0.9927 | 0.9810 | 0.9847 |
| 10 | 0.2085 | 0.3331 | 0.2580 |
| 11 | 0.9172 | 0.9177 | 0.9000 |
| 12 | 0.9173 | 0.7654 | 0.8195 |
| 13 | 0.6307 | 0.5604 | 0.5987 |
| 14 | 0.9916 | 0.9361 | 0.9671 |
| 15 | 0.9247 | 0.9172 | 0.9210 |
| 16 | 0.8160 | 0.7692 | 0.7362 |
| 17 | 0.9990 | 0.9917 | 0.9947 |
| 18 | 0.9795 | 0.8613 | 0.9360 |
| 19 | 0.1512 | 0.1593 | 0.1473 |
| 20 | 0.8201 | 0.7508 | 0.7533 |
| 21 | 0.7003 | 0.6167 | 0.6405 |
| 22 | 0.4322 | 0.3710 | 0.4282 |
| 23 | 0.9297 | 0.8783 | 0.8793 |
| 24 | 0.9303 | 0.8280 | 0.8396 |
| 25 | 0.5272 | 0.5540 | 0.5329 |
| 26 | 0.9908 | 0.9755 | 0.9705 |
| 27 | 0.9708 | 0.9104 | 0.9303 |

In practice, an investor should have this knowledge a priori. The values employed for our experiments are shown in Table 2. Using these values as a reference, we can initialize the fuzzy parameters for each investor profile. In the case of $\gamma$, we simply need to indicate a reference value between the highest and lowest value to indicate where the investor assigns a 0.5 level of satisfaction.

For the value of $\alpha$, we first need to indicate a reference value where the investor assigns the 0.99 level of satisfaction. Once $\gamma$ is defined, we use Equation 3 to isolate $\alpha$; this equation is used because of the generalized sigmoid function used within the GLCV:

$$\alpha = \frac{\ln(1/0.99) - 1}{x - \gamma}. \tag{10}$$

Here, $x$ denotes the reference value. It should be noted that while calculating $\alpha$ for the risk GLCV, we use $-\alpha$ to reflect its minimization. We selected the reference values for each profile based on a given percentage for each profile. The percentages used for each profile are shown in Table 3.

As shown in Table 3, all investors have an $\alpha$ of 99%. This indicates that every investor will have the maximum truth value on the highest return value and the lowest risk value. In the case of the aggressive investor, we see that the $\gamma$ point will be 90% close to the highest return value and 70% to the lowest return value.

This means that this investor will only assign values above 0.5 to the highest return portfolios while allowing the portfolio to have a higher risk. In contrast, the conservative investor will only assign values above 0.5 to those portfolios that are 90% close to the lowest risk while allowing the return to decrease.

### 5.1 Parameter Tuning

To assess the performance of the preference model, we ran a series of experiments with different PSO velocities and acceleration constants using a swarm of 50 particles for 120 iterations. The test values are shown in Table 4. Each combination of parameters was run 30 times. The average truth value of the predicate for each combination is shown in Table 5.

Table 5 identifies combination 17 as the best and combination 19 as the worst. Combination 17 has the values of velocity $= 5$, $p = 1.5$, and $g = 2$. Combination 19 has the values of velocity $= 1$, $p = 2$, and $g = 1$. In Figure 9, we can see the non-dominated solutions regarding combination 17.

**Table 6.** Average portfolio values

| Profile | Return | Risk |
|---|---|---|
| Markowitz function | 14.6413 | -4088.2392 |
| $I_a$ | 23.1744 | -4576.151 |
| $I_b$ | 22.1876 | -4664.703 |
| $I_c$ | 20.5939 | -4801.2728 |

**Table 7.** Reference portfolios

| | Return | Risk |
|---|---|---|
| High return | 23.2645 | -2424.8184 |
| Low risk | 14.471 | -5193.4265 |

**Table 8.** Average truth values

| | Degree of truth |
|---|---|
| A priori approach | 0.6383 |
| A posteriori approach | 0.4233 |

**Table 9.** Mann-Whitney-Wilcoxon results

| Variable | Value |
|---|---|
| $z$-score | 3.6794 |
| $u$-value | 5452 |
| $p$-value | 0.00024 |

Figure 9 shows how the profile solutions reflect the desired preference. The aggressive profile obtains solutions that have the most return and risk, while the conservative profile favors those solutions with lower return and risk, and the balanced profile is between the two.

These results are achieved due to the fuzzy parameters of Equation 8. The parameters set for the aggressive profile evaluate the solutions belonging to the conservative profile with a lower truth value and vice versa.

Only those solutions that align with the preference reflected by the fuzzy parameters will be assigned a high truth value. This guides the PSO to the desired Pareto-efficient solutions.

### 5.2 Comparative Analysis

In this section, we analyze the impact of the use of the CFL predicate during the optimization process. First, we set the PSO's parameters to those corresponding to the 17th combination. Then, we ran PSO 30 times using Equation 2 as the bi-objective function to approximate the whole Pareo frontier (a posteriori approach).

After that, we ran the PSO 30 times using Equation 8 for each investor profile to approximate the best compromise solution (a priori approach). The average return and risk of the best portfolios obtained in every execution are shown in Table 6.

From this set of solutions, we selected two new reference portfolios to create a second set of investor profiles. The values of these reference portfolios are shown in Table 7. Then, we reevaluated every portfolio using those reference points. Then, we evaluated every portfolio using both sets of investor profiles. Then, we reevaluated every portfolio using those reference points. The average truth value is shown in Table 8.

Lastly, we used the Mann-Whitney-Wilcoxon $U$ test to determine the statistical significance of the differences between the results from the a priori approach and the a posteriori one. The $U$ test is a non-parametric test to evaluate differences between two independent sets of values [13]. To determine if a difference exists or not, a given confidence level is employed, in this case 0.95.

Table 9 shows the test results. Given the 0.95 confidence level, a $p$-value below 0.05 indicates statistical differences between the two populations. In this case, this means that using the CFL predicate during the optimization process is more adequate to identify the solution that best matches the investor's preferences.

## 6 Conclusions and Future Research

In this paper, we proposed a preference model based on Compensatory Fuzzy Logic (CFL) to solve the Stock Portfolio Selection (PSS) problem. Using CFL, we were able to create predicates that expressed an investor's preferences regarding the return and risk of the desired portfolio.

These predicates are composed using sigmoid linguistic variables linked with CFL logical operators. The sigmoid function is selected because of its asymptotic properties that allow for a finer evaluation of a variable.

We integrated this model into a Particle Swarm Optimization (PSO) algorithm and ran a series of experiments with three investor profiles using data from the Mexican Stock Market. First, we did a parameter tuning experiment that revealed the best and worst PSO parameter values regarding the truth value of the CFL predicate.

The results suggest that the performance of the PSO is at its best when the swarm slightly favors global search with a medium velocity. In contrast, the worst performance corresponds to a swarm that has the minimum velocity and greatly favors local search.

Using the best combination of parameters, we see how the preference model consistently reflects the preference of each investor profile. It should also be noted that we in order to adjust to each investor profile, we only need to modify the parameters of the linguistic variables that form the CFL predicate.

We made a comparison based on the solutions obtained with PSO without the use of the CFL predicate during the optimization process and the solutions obtained by using CFL predicates. Investor profiles were created based on both results. Then, every solution was evaluated using both sets of investor profiles.

The results showed that the investor profiles obtained by using the CFL predicate during the optimization process have a higher truth value. The Mann-Whitney-Wilcoxon test showed that there were significant statistical differences between the two sets of solutions.

The main limitation of the model is that it needs to have information about the desired solution. Therefore, an investor unfamiliar with the possible outcome in terms of risk and return would not be able to initialize the CFL predicate properly.

For this reason, as future research, we propose the integration of an interactive module within the optimization process. This module should interact with the investor by presenting information about the found solutions and adjusting the sigmoid functions and the CFL predicate based on the investor's preferences. Another topic for future research is the use of another instance using indexes belonging to other markets. We also propose the implementation of the CFL predicate within a more complex heuristic, such as [11].

## References

1. **Abdolbaghi-Ataabadi, A., Nazemi, A., Saki, M. (2023).** Multi-objective possibility model for selecting the optimal stock portfolio. Advances in Mathematical Finance and Applications, Vol. 8, No. 2. DOI: 10.22034/AMFA.2022.1952682.1705.
2. **Amiri, M., Heidary, M. S. (2019).** Portfolio optimization with robust possibilistic programming. Iranian Journal of Finance, Vol. 3, No. 2, pp. 44–65. DOI: doi.org/10.22034/ijf.2020.195328.1046.
3. **Corazza, M., di-Tollo, G., Fasano, G., Pesenti, R. (2021).** A novel hybrid PSO-based metaheuristic for costly portfolio selection problems. Annals of Operations Research, Vol. 304, No. 1–2, pp. 109–137. DOI: 10.1007/s10479-021-04075-3.
4. **Dai, Y., Qin, Z. (2021).** Multi-period uncertain portfolio optimization model with minimum transaction lots and dynamic risk preference. Applied Soft Computing, Vol. 109, pp. 107519. DOI: 10.1016/j.asoc.2021.107519.
5. **Doering, J., Kizys, R., Juan, A. A., Fito, A., Polat, O. (2019).** Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends. Operations Research Perspectives, Vol. 6, pp. 100121. DOI: 10.1016/j.orp.2019.100121.
6. **Espin-Andrade, R. A., Cruz-Reyes, L., Llorente-Peralta, C., Gonzalez-Caballero, E., Pedrycz, W., Ruiz, S. (2021).** Archimedean compensatory fuzzy logic as a pluralist contextual theory useful for knowledge discovery. International Journal of Fuzzy Systems, Vol. 24, No. 1, pp. 1–21. DOI: 10.1007/s40815-021-01150-6.

7. **Espin-Andrade, R. A., González-Caballero, E., Pedrycz, W., Fernández-González, E. R. (2016).** Archimedean-compensatory fuzzy logic systems. International Journal of Computational Intelligence Systems, Vol. 8, No. Supplement 2, pp. 54–62. DOI: 10.1080/18756891.2015.1129591.
8. **Fazli, M., Lashkari, M., Taherkhani, H., Habibi, J. (2022).** A novel experts advice aggregation framework using deep reinforcement learning for portfolio management. arXiv. DOI: 10.48550/arXiv.2212.14477.
9. **Hamdi, A., Karimi, A., Mehrdoust, F., Belhaouari, S. B. (2022).** Portfolio selection problem using CVaR risk measures equipped with DEA, PSO, and ICA algorithms. Mathematics, Vol. 10, No. 15, pp. 2808. DOI: 10.3390/math10152808.
10. **Harris, R. D., Mazibas, M. (2022).** Portfolio optimization with behavioural preferences and investor memory. European Journal of Operational Research, Vol. 296, No. 1, pp. 368–387. DOI: 10.1016/j.ejor.2021.04.044.
11. **Kawano, Y., Valdez, F., Castillo, O. (2022).** Fuzzy combination of moth-flame optimization and lightning search algorithm with fuzzy dynamic parameter adjustment. Computación y Sistemas, Vol. 26, No. 2, pp. 743–757. DOI: 10.13053/cys-26-2-4269.
12. **Li, B., Shu, Y., Sun, Y., Teo, K. L. (2021).** An optimistic value–variance–entropy model of uncertain portfolio optimization problem under different risk preferences. Soft Computing, Vol. 25, No. 5, pp. 3993–4001. DOI: 10.1007/s00500-020-05423-4.
13. **Mann, H. B., Whitney, D. R. (1947).** On a test of whether one of two random variables is stochastically larger than the other. The Annals of Mathematical Statistics, Vol. 18, No. 1, pp. 50–60. DOI: 10.1214/aoms/1177730491.
14. **Markowitz, H. M. (1952).** Portfolio selection. The journal of Finance, Vol. 7, No. 1, pp. 71–91. DOI: 10.2307/2975974.
15. **Nozarpour, Y., Davoodi, S. M. R., Fadaee, M. (2023).** Selecting the optimal multi-period stock portfolio with different time horizons in the credibility theory framework. Advances in Mathematical Finance and Applications, Vol. 8, No. 3, pp. 1043–1056. DOI: 10.22034/amfa.2022.1953564.1709.
16. **Thakur, G. S. M., Bhattacharyya, R., Sarkar, S. (2022).** Fuzzy expert system for stock portfolio selection: An application to bombay stock exchange. arXiv. DOI: 10.48550/arXiv.2204.13385.
17. **Zadeh, L. A. (1965).** Fuzzy sets. Information and Control, Vol. 8, No. 3, pp. 338–353. DOI: 10.1016/S0019-9958(65)90241-X.

*Article received on 28/02/2024; accepted on 14/05/2024.*
**Corresponding author is Gilberto Rivera.*

ISSN 2007-9737

# Spanish Automatic Text Summarization: A Survey

Griselda Areli Matías-Mendoza, Yulia Ledeneva*, René Arnulfo García-Hernández

Autonomous University of the State of Mexico,
Mexico

{gris_9123, renearnulfo}@hotmail.com, yledeneva@yahoo.com

**Abstract.** The exponential growth of textual data has necessitated efficient summarization techniques. However, it is difficult for humans to summarize large text documents manually. As a result, automatic text summarization has emerged as a crucial and effective tool for helping to interpret and manage text information. Given the limited time available to read and fully comprehend each document before making decisions, there is a strong need for summarizing documents to convey a clear, representative idea of the original content. This has important practical applications in information retrieval, document classification, and knowledge extraction. Moreover, advanced summarization systems can effectively identify the core ideas of texts, significantly reducing the time users spend reading entire documents. While automatic text summarization has been extensively researched for languages, such as English, over the last 60 years, Spanish has received less attention. This paper addresses this gap by presenting key approaches, challenges, and methodologies in Spanish automatic text summarization. Through a comprehensive survey of relevant literature, we aim to provide a foundation for future research in this area. The presented survey is a compilation of important works in Spanish automatic text summarization and is intended to be a basis for research in the task. Also, we determine the main challenges for the task of Spanish automatic text summarization.

**Keywords.** Automatic text summarization, state-of-the-art methods for Spanish, abstractive text summarization, document summarization, news summarization, extractive summarization, natural language processing, corpus TER, corpus DUC, corpus TAC, evaluation of text summaries.

## 1 Introduction

Summaries are ubiquitous in our daily lives, from books and news articles to films, audio, scientific papers, and even social media platforms like Twitter. A summary can be defined as a condensed version of one or more texts that highlights key information while maintaining a length typically less than half of the original [1]. While traditionally applied to text, automatic summarization can also be used for other media, such as video and audio.

The explosion of information online has created a demand for tools that can quickly and efficiently extract key points from vast amounts of text. Automatic text summarization research has been ongoing since the 1950s, with Luhn's pioneering work in 1958 [2]. Over the decades, researchers have continually refined techniques to produce summaries that resemble those created by humans.

A summary can be generated through extractive, abstraction, and hybrid methods. Abstractive methods involve a complex process that requires significant computational resources and advanced linguistic techniques. Extractive methods create summaries by selecting and extracting the most important text elements, such as sentences, phrases, or paragraphs. The hybrid methods combine extractive and abstraction methods. The research community focuses more on extractive summaries, achieving more coherent and meaningful summaries [3].

The state-of-the-art methods for generating summaries also take into account the distribution of sentences and structure to identify and extract the most important ones [4-9]. These methods also use the text model to maintain the consistency of the summaries [10-13]. Another significant problem is the need for an equitable study of this task for different languages.

For example, before 2000, research on automatic text summarization primarily focused on

English because resources such as standard evaluation measures and corpora were available for this language. Despite this, other languages, like Spanish, have shown substantial growth. Spanish is now the world's second most spoken language and the third most used online, as noted in [14].

This creates an excellent prospect for advance study in Spanish automatic text summarization. This area has been the need for gold-standard summaries in Spanish. However, this is starting to improve, especially with the inclusion of the Spanish language in the corpora and tasks of the ACL 2013 MultiLing Workshop.

While other surveys and reviews cover general automatic text summarization, this one specifically examines Spanish-language summarization. It offers a comprehensive overview of existing research. Additionally, it covers the methods used in Spanish automatic text summarization, evaluates the outcomes, and presents relevant corpora, conferences, and workshops. The survey also addresses the most significant challenges in the area and completes with recommendations and suggestions for future research.

## 2 Natural Language Processing for Spanish Language

In 2023, over 599 million people spoke Spanish as their native language. Additionally, the number of potential Spanish users worldwide exceeds 585 million. Spanish ranks as the second most spoken native language globally, following Mandarin Chinese, and is also the second most spoken language overall when considering native speakers, those with limited proficiency, and Spanish learners.

Regarding institutional recognition, Spanish holds the third position as a working language within the United Nations and ranks fourth within the European Union. Spanish is the third most widely used language online, especially on platforms like Wikipedia, Facebook, and Twitter, where it holds second place in usage [14].

Spanish is said to come from the Romance languages, which do not derive from the Latin written in literature but from the Latin spoken in the streets and places [15]. While its roots trace back to the 3rd century A.C., its distinct development occurred centuries later.

Spanish is spoken in almost all the Iberian Peninsula, in the southwest of the U.S.A., throughout Mexico, and in Central and South America (except for Brazil and Guayana). In addition, it is the language of a minority group of speakers in the Philippines. This vast geographical spread brings, consequently, a significant range of dialectal variants.

However, despite being a language spoken in such distant areas, there is a certain uniformity in the cultured level of the language that allows people on either side of the Atlantic to understand each other relatively quickly. The most significant differences are suprasegmental, that is, the varied intonation, apparently the result of the different linguistic substrates in Spanish-speaking countries.

The Spanish language is composed of 26 letters of the Latin alphabet. Like Spanish, languages such as English (universal language), Portuguese, German, French, Swedish, and others use the Latin alphabet, so it is not difficult to become familiar with its symbology since it is not as complex as in languages such as Arabic or Russian. Currently, the universal language of world communication is English, so most of the research in the different areas of Natural Language Processing (PLN) has been carried out in this language, especially automatic text summarization.

One of the problems between languages is that specific characteristics depend on each language and simplify or make the relationship between groups of words more complete. However, English and Spanish use the same alphabet and have a basic order in the composition of their sentences: subject + verb + complement; this does not mean that this order is always fulfilled.

English has a stricter order, which must be conserved. However, the Spanish had more freedom, for example (see Table 1). The freedom of the Spanish language to create sentences complicates the automatic abstractive text summarization task. However, automatic extractive text summarization is a task very similar to that performed in English due to the use of the same alphabet and the coincidence between the composition of the sentences.

**Table 1.** Example of the composition of sentences in Spanish [16]

| Example | Structure |
|---|---|
| *Juan vino a mi casa* | Subject + Verb + Complement |
| *A mi casa vino Juan* | Complement + Verb + Subject |
| *Vino Juan a mi casa* | Verb + Subject + Complement |
| *A mi casa Juan vino* | Complement + Subject + Verb |
| *Juan a mi casa vino* | Subject + Complement + Verb |
| *Vino a mi casa Juan* | Verb + Complement + Subject |

## 3 History of Text Summarization: Corpus and Evaluation

Automatic text summarization has been the research subject for 60 years, beginning in the 1950s with Luhn's pioneering work in 1958 [17]. Luhn was the first to apply automatic extractive text summarization using text similarity. Later, in 1969, Edmundson introduced features such as word frequency, sentence position, title, and pragmatic words, which are still relevant and utilized today [18].

The advance of automatic text summarization in the following years was stopped, and only some investigations were carried out, such as those of Rush et al.'s work in 1971-1975 [19, 20] and Gerald Francis DeJong's studies in 1982 [21]. In 1993, research took off again with work by Spärck-Jones [22] and 1995 Julia Kupiec et al. [23]. This research helped to revive an interest in studying automatic text summarization.

Among the works that followed are [24-27]. Until 2000, most research in automatic text summarization focused exclusively on the English language. It was conducted without a standard corpus or evaluation measures, making comparison across studies difficult.

For example, the research in [17] used 50 journalistic articles, [18] utilized 200 articles, [28] analyzed 30 documents, [23] examined 188 scientific documents, and [29] worked with 30 documents. In 2001, the Document Understanding Conferences (DUC) were established to promote progress in summarization for English and provide a large-scale platform for researchers. DUC consisted of seven conferences: DUC01 through DUC07. Each conference included several tasks, with a corresponding gold standard corpus developed for each task.

Building on the foundation laid by the DUC conferences, the Text Analysis Conference (TAC) emerged in 2008 as a significant player in automatic text summarization. TAC's workshops were designed to elevate system evaluation, focusing on multi-document summaries for end-users. The TAC corpus, which concentrated on summaries produced between 2008 and 2014, is a testament to TAC's commitment to advancing the field. Table 3 provides an overview of the TAC corpora, further highlighting its role in the field.

In 2011, the MultiLing task was introduced to evaluate language-independent summarization algorithms across different languages. MultiLing corpora were produced in 2011, 2013, 2015, and 2017 for multilingual automatic text summarization. While MultiLing includes multiple languages, the original texts are primarily in English and translated into various languages, so there is no native corpus for each language [30-32]. Table 2 presents the standard datasets for text summarization.

Despite existing research on Spanish, a standardized or specialized corpus is essential for developing effective automatic text summarization systems. Many researchers have adapted corpora from information extraction tasks or created their own for Spanish automatic text summarization [58- 64].

This inconsistency hinders direct comparisons and makes it difficult to assess the progress in this field. To address this issue, recent efforts have focused on developing a standardized Spanish corpus. The CNN corpus was created in 2019, with the Spanish version based on news articles sourced from the CNN Mexico website. These articles address various general-interest topics and are written in standard language.

The corpus features summaries written by the original authors in English, emphasizing the key points of the CNN texts. It also includes the original text, story highlights, and additional metadata such as author names, titles, subject classifications, and publication dates, all retrieved from the Spanish

**Table 2.** Overview of existing corpora for summarization

| Corpus | Lang. | Domain | Single-Doc. | Multi-Doc | Size |
|---|---|---|---|---|---|
| DUC 2001 [33] | English | News | Yes | Yes | 60 x 10 |
| DUC 2002 [34] | English | News | Yes | Yes | 60 x 10 |
| DUC 2003 [35] | English | News | Yes | Yes | 60 x 10, 30 x 25 |
| DUC 2004 [36] | English/Arabic | News | Yes | Yes | 100x10 |
| DUC 2005 [37] | English | News | | Yes | 50 x 32 |
| DUC 2006 [38] | English | News | | Yes | 50 x 25 |
| DUC 2007 [39] | English | News | | Yes | 25 x 10 |
| TAC 2008 [40] | English | News | | Yes | 48 x 20 |
| TAC 2009 [41] | English | News | | Yes | 44 x 20 |
| TAC 2010 [42] | English | News | | Yes | 46 x 20 |
| TAC 2011 [43] | English | News | | Yes | 44 x 20 |
| ICSI [44] | English | Meetings | Yes | | 57 |
| AMI [45] | English | Meetings | Yes | | 137 |
| Opinosis [46] | English | Reviews | Yes | Yes | 51 x 100 |
| Gigaword [47] | English | News | Yes | | 4,111,240 |
| Gigaword 5 [48] | English | News | Yes | | 9,876,086 |
| LCSTS [49] | Chinese | blogs | Yes | | 2,400,591 |
| CNN/Daily Mail [50] | English | News | Yes | | 312,084 |
| MSR Abstractive [51] | English | misc | Yes | | 6,000 |
| arXiv [52] | English | science | Yes | | 194,000 |
| PubMed [52] | English | science | Yes | | 278,000 |
| EASC [53] | Arabic | News/Wikipedia | Yes | | 153 |
| SummBank [54] | Chinese/English | News | Yes | Yes | 40 x 10 |
| CAST [55] | English | News | Yes | | 147 |
| CNN-corpus [56] | English | News | Yes | | 3,000 |
| TeMário [57] | Portugues | News | Yes | | 100 |

version of the CNN website. The development of the Spanish CNN corpus followed the methodology proposed by Lins et al. in 2019.

In 2020, the TER standard corpus for Automatic Text Summarization in Spanish was created. TER is a corpus of Mexican Spanish-language news from the “Crónica” newspaper.

The construction of the corpus is divided into two stages: the first for the selection, cleaning, and tagging of news, and the second for the selection of experts, construction, and tagging of summaries [66].

In addition, a Corpus, composed of documents from various languages, has been generated, such as Multilingual Summarization Corpus (MLSUM).MLSUM is the first extensive dataset of its kind, featuring over 1.5 million article-summary pairs across five languages: Turkish, Spanish, Russian German, and French. Sourced from online newspapers, this valuable resource is a

**Table 3.** Overview of existing corpora for summarization in Spanish

| Corpus | Lang. | Domain | Single-Document | Multi-Document | Size |
|---|---|---|---|---|---|
| ABC | Spanish | News | Yes | | 109 |
| Medical articles | Spanish | Science | Yes | | 20 |
| Desastres | Spanish | News | | Yes | 300 |
| CNN-Corpus Spanish | Spanish | News | Yes | | 1117 |
| TER | Spanish | News | Yes | | 240 |
| MLSUM | Spanish | News | Yes | | 290,645 |
| DACSA | Spanish | News | Yes | | 2,120,649 |
| Bernoldi | Spanish | News | Yes | | 93,913 |

cornerstone for advancing multilingual summarization research.

For the Spanish language, the newspaper El País was used in that article [67]. Segarra et al.'s research describes the construction of a corpus of Catalan and Spanish newspapers, the Dataset for Automatic Summary of Catalan and Spanish period Articles (DACSA).

It is a large-scale, high-quality corpus that can be used to train summary models for Catalan and Spanish [68]. In [69], a corpus is built from the website of the Spanish newspaper “20 Minutos”, which has a history of news that is freely accessible and downloadable. This corpus's main objective is to generate abstract summaries of news in Spanish automatically. Table 3 provides a brief description of the corpora for summary in Spanish.

Standard construction data (corpus) and various evaluation methods are necessary to assess automatically generated summaries. These evaluation methods are divided into intrinsic and extrinsic categories [70]. Intrinsic methods directly analyze the automatically produced summary, evaluating grammatical correctness, cohesion, and coherence to determine its quality.

These methods typically compare automatically generated summaries with expert-created ones to evaluate coverage. On the other hand, extrinsic evaluation methods assess the summary in the context of the task for which it was created, aiming to measure its impact on the performance of related tasks. These tasks may include, for example, relevance evaluation [71].

The most widely used evaluation method in automatic text summarization is ROUGE (Recall-Oriented Understudy for Gisting Evaluation), introduced by Lin and Hovy [72], [73]. ROUGE compares system-generated summaries with human-created (gold standard) summaries using n-gram statistics. ROUGE offers several automatic evaluation metrics for this purpose:

– ROUGE-N (n-grams co-occurrence).

This metric measures the recall or coverage of n-grams between a candidate summary and a set of reference summaries. It is calculated using the following formula (Formula 1):

$$ROUGE-N = \frac{\sum_{set\{PeerSummary\}}\sum_{gram_n \epsilon S} Count_{match}(gram_n)}{\sum_{S\epsilon\{PeerSummary\}}\sum_{gram_n \epsilon S} Count(gram_n)}, \quad (1)$$

where $n$ is the length of the n-gram and $Count_{match}(gram_n)$ is the maximal number of n-grams that co-occur in the candidate summary and in the set of reference summaries.

ROUGE-N evaluates the quality of candidate summaries by quantifying the overlap of n-grams between the candidate and reference summaries. The score ranges from 0 to 1, where 0 signifies no overlap between the candidate and reference texts, while 1 indicates a complete overlap. ROUGE-N helps determine how well a system captures key content and linguistic details.

This metric, which evaluates the occurrence of noncontiguous bigrams, is a crucial component in automatic text summarization. Noncontiguous bigrams are any two words that appear in the same order within a sentence, regardless of the number of intervening words. The co-occurrence of noncontiguous bigrams provides a statistical

**Table 4.** Summary of survey

| Name | Language |
|---|---|
| A Survey for Multi-Document Summarization [77] | English |
| A Survey on Automatic Text Summarization [78] | English |
| A Comprehensive Survey on Text Summarization Systems [79] | English |
| A Survey of Text Summarization Extractive Techniques [80] | English |
| Query-Based Summarization: A survey [81] | English |
| A Survey of Text Summarization Techniques [82] | English |
| A Survey of Unstructured Text Summarization Techniques [83] | English |
| A Survey on Automatic Text Summarization [84] | English |
| Automatic Arabic text summarization: a survey [85] | Arabic |
| Recent automatic text summarization techniques: a survey [86] | English |
| Automatic Arabic Summarization: A survey of methodologies and systems [87] | Arabic |
| Text Summarization Techniques: A Brief Survey [88] | English |

measure of how well the candidate summary captures the noncontiguous bigrams from the reference summaries. Lin [72] demonstrated that this measure can effectively assess the quality of automatically generated summaries, achieving a 95% correlation with human judgments.

Since the introduction of standard corpora, automatic text summarization has gained importance, leading to over 400 studies focusing on the English language.

Few studies have focused on researching automatic text summarization for the Spanish language. In 2001, Acero et al. [58] presented the automatic generation of personalized summaries using their corpus, built from news articles from the ABC newspaper. Villatoro [61] used a similar corpus to extract and adapt information for automatic multi-document summarization in Spanish [74]. Other studies related to Spanish automatic summarization include [58-59], [61-62], [64], and [75-76].

However, despite these efforts, progress remains unclear because researchers have used either custom or adapted corpora, which prevents consistent comparisons between different methods. While a standard corpus exists, many state-of-the-art techniques have not yet been tested to evaluate their performance. In recent years, there has been growing interest in compiling research on automatic text summarization across various languages. Table 4 provides a list of different surveys conducted in this field. However, we still need an overview of the study of automatic text summarization for the Spanish language.

## 4 Spanish Automatic Text Summarization Approaches

Several generic automatic text summarization algorithms have been developed, each with advantages and disadvantages and different classifications depending on the technique or the input type. This section presents a survey of the literature on Spanish automatic text summarization. Due to the few Spanish automatic text summarization investigations, each state-of-the-art method that works with Spanish is described.

– Automatic Generation of Personalized Summaries [58]. This work is a practical application within Hermes, a personalized news dispatcher that handles information in English and Spanish. This system effectively utilizes three heuristics to select phrases to realize the summary.
  1 Sentence position heuristic. It consists of giving a higher score to the first five sentences of a text.
  2 Keyword heuristic. It consists of extracting the M most significant words from each

text and then checking how many of these keywords are found in each phrase. In this way, the highest number of phrases with the highest number of keywords is assigned.

3 Personalization heuristic. It consists of promoting phrases most relevant for a user model to personalize the summary.

The corpus consists of 109 news obtained in the electronic edition of the newspaper ABC.

– Towards a Linguistic Model of Automatic Summary of Medical Articles in Span-ish [60]. It focuses on the specialized Spanish automatic text summarization, specifically in medicine. The corpus he uses consists of 20 medical articles in Spanish that are part of the Technical Corpus of the Institut Universitari de Lingüística Aplicada (IULA) of the Fabra University of Barcelona. The method that is used consists of four stages.

1 Selection of work corpus. The selected corpus is divided into two subcorpus, reference and contrast.

2 Analysis of the texts of the reference subcorpus. The text structure of the medical article, its representative lexical units, and its discursive, syntactic, and communicative structure are analyzed.

3 Development of the model.

- Definition of the summary model.
- Development of linguistic rules.
- Manual validation of the operation of the rules.
- Implementation of the rules.
- Application of the rules on texts of the contrast subcorpus.

4 Evaluation of the model.

– Approach to the Automatic Summary as a tool to help legal translation in the field of tourism law [59]. This research is done for documents in Spanish in the tourism law field. However, it does not present any method for automatic text summarization since it only applies to the Copernic Summarizer tool to generate the summaries that later serve to translate.

– The Platform for Language Independent Summarization [64] introduces a summarization platform that operates independently of language. It supports tasks such as corpus acquisition, language classification, translation, and text summarization across 25 different languages. When the input text is in English, it is processed by an automatic extractive summarization module. This module selects the most important sentences from the original text using well-established sentence scoring methods, known for their high efficiency in extractive summarization. For texts in other languages, the platform employs language-independent summarization algorithms, and various translation tools are used to convert the sentences into English. Since automatic translation may cause some semantic loss, utilizing multiple translation tools can help mitigate these issues. The resulting translated versions are then fed into the extractive summarization module, where each version generates scores for the sentences in relation to the original text. The Sentence Scoring and Selection Module evaluates the chosen sentence sets and produces a final summary by selecting the corresponding sentences from the original text.

The corpus used in this platform is CNN-Spanish, with the current version containing 400 texts classified into eight categories: sports, entertainment, world, national, opinion, technology, travel, and health news.

– Automatic Summarization of Multiple Documents [61]. Villatoro's work utilizes a classifier and supervised learning tools. The core concept is that an inductive process automatically builds a classifier by analyzing the characteristics of a set of previously summarized documents. The learning algorithm receives pairs of (documents and summaries), turning the task of generating summaries into a supervised learning process. The Disaster dataset was used for experimentation with Spanish-language corpora [89]. Although the corpus was originally designed for classification, it was adapted for automatic text summarization. The

Disaster dataset consists of 300 news articles collected from Mexican newspapers. Each sentence was labeled with two tags: Relevant and Non-Relevant. To minimize subjectivity in the labeling process, experts were instructed to label a sentence as "Relevant" only if it contained at least one factual detail about the event, such as the date, location, the number of affected people or homes, economic damages, or the scale or magnitude of the disaster.

– Automatic Generation of Summaries [90]. A method based on supervised learning techniques is proposed, specifically in classification. The corpus he uses is composed of more than 8000 documents containing nine years of rectoral resolutions of the Catholic University of Salta. The method uses a labeling process to determine whether sentences are relevant. In addition, each sentence must have a label that indicates whether it belongs to the summary. They used the We-ka software tool for the experiments, which included a vast collection of classification techniques. Among the classifiers this method uses are ADTree, ID3, C4.5 with pruning, C4.5 without pruning, Decision Table, Ripper, and Naïve Bayes. The construction of decision trees obtains summaries of adequate quality, which serve as indicative summaries for the user of a semantic search engine in the proposed corpus in this research.
– A New Cross-Lingua Automatic Summarization Approach Based on Textual Energy [91]. This method introduces a cross-language summarization system that incorporates textual energy and translation time measurement, improving the reliability of the final news summaries. The automatic summarization technique, which uses textual energy, is inspired by statistical physics and combines a Vector Space Model (VSM) with neural networks. The ENERTEX method [92] treats words in the text as units that interact and are influenced by the field generated by each unit. As a result, each word is assigned a score based on its textual energy. Additionally, this approach factors in the translation time of each sentence. A textual energy matrix is generated, aiding in the summary creation process. The system's performance was evaluated using the FRESA framework, which compared the automatically generated summaries with baseline summaries for varying percentages of the original texts.
– PuertoTex: A Data Mining Software Based on Ontologies for Automatic Summarization in the Port and Coastal Engineering Domain [93]. This research focuses on developing and evaluating an ontology-based software designed to automatically generate summaries in the field of Ports and Coastal Engineering. The tool's development incorporates techniques from discourse analysis and cognitive methods to create rules for processing texts. It also involves constructing an ontology to support labeling processes, utilizing the capabilities of the Resource Description Framework and Extensible Markup Language. A set of agents was created to act on the ontology, defining its essential elements. The resulting product is the PuertoTex software, which generates ontology-based automatic summaries. This method was tested in both English and Spanish. Three evaluation approaches were employed: usability evaluation, information retrieval evaluation, and an assessment of the automatically generated summary.
– Automatic Sentence Compression: a Study towards the Generation of Summaries in Spanish [76]. This research explores sentence compression techniques for Spanish summarization. A linear model that predicts the removal of intra-sentence segments based on a set of text-based features were proposed. The model was trained on a large dataset of over 60,000 sentences, considering the entire context and the generated summary. Through statistical analysis, the most significant features for predicting segment deletion with 75% accuracy were identified. Then, two algorithms are proposed for generating summaries with compressed sentences after summaries are evaluated with a test similar to the Turing Test.
– Automatic Generation of Summaries with Support in Ontologies Applied to the Biomedical Domain [94]. This research

proposes an architecture for generating informative summaries of a single document in a specific domain: biomedicine. A method of extracting sentences is presented, based on the theory of complex networks, which maps the text to the concepts of the UMLS ontology and represents the document and the sentences as graphs. The selection of sentences is based on the degree of connection of their concepts in the graph of the document, using a grouping algorithm based on connectivity. A system that implements the proposed method is developed, and the empirical results of applying different heuristics to select the summary sentences are shown.

– Evaluation of Summaries in Spanish with Latent Semantic Analysis: A Possible Implementation [63]. This research seeks to identify an effective method for evaluating summaries using Latent Semantic Analysis (LSA). Secondary school students from Valparaíso, Chile, wrote the summaries. To achieve this goal, the scores assigned by three teachers to 244 summaries of primarily expository texts and 129 summaries of mostly narrative texts were compared with the scores produced by three computational methods based on LSA. The methods include:

1 Comparison of summaries with the source text.
2 Comparison of summaries with a summary developed by the consensus of a group of linguists.
3 Comparison of summaries with three summaries constructed by three language teachers.

– Text Summarization of Spanish Documents [95]. This research aimed to develop an extraction-based automatic text summarization algorithm. The proposed method involves constructing a directed weighted graph from the original text. A ranking algorithm is then applied to identify the most important sentences based on the weighted graph, ensuring that these critical sentences are included in the summary. The project's primary objective was to summarize 642 news articles computationally while ensuring no essential information was omitted from the summaries.

– Ground Truth Spanish Automatic Extractive Text Summarization Bounds [66]. This research introduces the TER standard corpus, designed to evaluate state-of-the-art methods and systems for automatic summarization in the Spanish language. The essential contribution lies in proposing the configuration and evaluation of five state-of-the-art methods, five systems, and four heuristics using three evaluation metrics: ROUGE, ROUGE-C, and Jensen-Shannon divergence. Notably, this study marks the first use of Jensen-Shannon divergence to assess automatic summarization in Spanish. In Matias (2020), ground truth bounds for Spanish were presented, including the heuristic baselines of first, random, topline, and concordance. Additionally, a ranking of 30 evaluation tests for state-of-the-art methods and systems was established, creating a benchmark for automatic summarization in Spanish.

– Evaluating Extractive Automatic Text Summarization Techniques in Spanish [96]. This study assesses both traditional and innovative extractive text summarization techniques in Spanish. The Corpus-TER [66], a dataset compiled from Mexican-Spanish news websites, was used for this evaluation. The primary objectives of the research are:

1. Select and develop specific summarization methods,
2. Choose a suitable corpus for testing these methods,
3. Design a concise and reusable interface and
4. Evaluate the summarization techniques.

The evaluation process utilizes the ROUGE and BLUE tools to assess performance.

– Generación Automática de Resúmenes Abstractivos de Noticias en Español [69]. In this work, we propose and evaluate a BERT-based processing pipeline for generating abstractive summaries of Spanish news. Specifically, it uses the BERTSUM framework on BETO [98], a model pre-trained exclusively in Spanish. On this basis, the model parameters are adjusted with a corpus of Spanish news. The work evaluates its results using the ROUGE metric and compares them

with some results obtained in English with the CNN/Daily Mail corpus.

– esT5s: A Spanish Model for Text Summarization [99]. The paper is about building a deep learning model for the task of Spanish text summarization based on the T5 (Text-to-Text Transfer Transformer) architecture. Such models have made significant progress in natural language processing, especially in English, but Spanish and other languages require specific models, the training of which is often computationally expensive. The work described in the paper addresses building a Spanish text summarization model from a large multilingual model, in this case, the mT5 model, which includes 101 languages. The authors managed to create a specialized model for Spanish called esT5, which is more efficient in terms of training time and computational power required. This model can be trained in less than an hour using a single GPU and produces summaries of comparable quality to larger models, significantly faster at inference.
– XL-Sum: Introduces a large-scale multilingual dataset designed for automatic abstractive summarization. This dataset includes over one million article-abstract pairs in 44 languages, including Spanish. The dataset was collected from BBC news articles using an automated process that extracts professional summaries written by human authors. It is highlighted that the dataset includes summaries in Spanish, which is significant due to the scarcity of high-quality public datasets in this language for abstractive summarization tasks.

## 5 Discussion

In the previous sections, several research studies on automatic text summarization were addressed, first general and later focused on the Spanish language. The main objective was to present a general overview of the task to understand the Span-ish automatic text summarization problem. While there are more than 400 studies for the English language and various studies on automatic text summarization, less than 24 research are available for the Spanish language.

The investigations in Spanish for automatic text summarization cannot be compared because each works with different corpora and various objectives. Even though the Spanish automatic text summarization research is approximately 20 years old, there has yet to be much progress; this is likely because Spanish did not hold significant global importance or was not extensively utilized. However, due to the growth of native and foreign speakers, and above all, on the Internet, automatic text summarization in Spanish has become essential.

In recent years, state-of-the-art methods began to present language independence [61], [100-104]; however, they have been tested in other languages, such as English, Arabic, and Portuguese, but not in Spanish. This is mainly due to the need for a standard corpus.

The nature of the Spanish language is very similar to that of English. English is the most studied language in automatic text summarization, so state-of-the-art methods of automatically generating summaries, mainly extractive and multilanguage, are created and tested in English. However, applying these methods to the Spanish language would be possible due to the language's nature.

There is no investigation into automatic abstractive text summarization for the Spanish language. Moreover, most of the investigations carried out are for extractive summaries of a single document; only one of those presented is for multiple documents. Therefore, this represents a great research opportunity for Spanish automatic text summarization.

The evaluation methods proposed for the English language [72] can be used since most of them are based on the correlation between the words of the automatically generated summary and the gold standard (made by the human).

## 6 Conclusion

This paper provides a comprehensive overview of the existing literature on Spanish automatic text summarization. We explore a range of methods used for both summary generation and evaluation, highlighting the relatively recent and understudied nature of this research area.

To advance Spanish automatic text summarization, future studies should consider adapting state-of-the-art methods from English and exploring related research in the field of natural language processing. A significant challenge in Spanish summarization is the lack of high-quality gold-standard summaries. Addressing this issue through the creation of a standardized corpus would enable researchers to test existing extractive summarization methods and fine-tune their parameters for Spanish.

Subsequently, the parameters of the methods for the Spanish language can be adjusted. There is a large field of research in generating automatic abstractive text summarization.

Finally, the development of automatic abstractive summarization systems for Spanish remains a promising area for future research.

## References

1. **Radev, D. R., Hovy, E., McKeown, K. (2002).** Introduction to the special issue on summarization. Computational linguistics, Vol. 28, No. 4, pp. 399–408.
2. **Luhn, H. P. (1958).** The automatic creation of literature abstracts. IBM Journal of research and development, Vol. 2, No. 2, pp. 159–165. DOI: 10.1147/rd.22.0159.
3. **El-Kassas, W. S., Salama, C. R., Rafea, A. A., Mohamed, H. K. (2021).** Automatic text summarization: A comprehensive survey. Expert Systems with Applications, Vol. 165, p. 113679. DOI: 10.1016/j.eswa.2020.113679.
4. **Mendoza, M., Bonilla, S., Noguera, C., Cobos, C., León, E. (2014).** Extractive single-document summarization based on genetic operators and guided local search. Expert Systems with Applications, Vol. 41, No. 9, pp. 4158–4169. DOI: 10.1016/j.eswa.2013.12.042.
5. **Nandhini, K., Balasundaram, S. R. (2014).** Extracting easy to understand summary using differential evolution algorithm. Swarm and Evolutionary Computation, Vol. 16, pp. 19–27. DOI: 10.1016/j.swevo.2013.12.004.
6. **Qazvinian, V., Hassanabadi, L. S., Halavati, R. (2008).** Summarizing text with a genetic algorithm-based sentence extraction. International Journal of Knowledge Management Studies, Vol. 2, No. 4, pp. 426–444. DOI: 0.1504/IJKMS.2008.01975.
7. **Mateo, P. L., González, J. C., Villena, J., Martínez, J. L. (2003).** Un sistema para resumen automático de textos en castellano. Procesamiento del lenguaje natural, Vol. 31, pp. 29–36.
8. **Babar, S., Patil, P. D. (2015).** Improving performance of text summarization. Procedia Computer Science, Vol. 46, pp. 354–363. DOI: 10.1016/j.procs.2015.02.031.
9. **Kiyoumarsi, F. (2015).** Evaluation of automatic text summarizations based on human summaries. Procedia-Social and Behavioral Sciences, Vol. 192, pp. 83–91. DOI: 10.1016/j.sbspro.2015.06.013.
10. **Sidorov, G. (2019).** Syntactic n-grams in computational linguistics. Cham, Switzerland: Springer International Publishing. DOI: 10.1007/978-3-030-14771-6.
11. **Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L. (2014).** Syntactic n-grams as machine learning features for natural language processing. Expert Systems with Applications, Vol. 41, No. 3, pp. 853–860. DOI: 10.1016/j.eswa.2013.08.015.
12. **Ledeneva, Y., García-Hernández, R. A. (2017).** Automatic generation of text summaries. Challenges, Proposals and Experiments, Universidad Autónoma del Estado de México, Toluca.
13. **Matias, G., Ledeneva, Y., García Hernández, R. A. (2020).** Detección de ideas principales y composición de resúmenes en inglés, español, portugués y ruso. 60 años de investigación (Alfaomega - UAEMex.). Alfaomega Grupo Editor, SA de CV.
14. **Vítores, D. F., Cervantes, I. (2023).** El español: una lengua viva. Informe 2023. El español en el mundo 2023: Anuario del Instituto Cervantes, pp. 23–142.
15. **Huidobro, J. M. (2016).** Origen y evolución del castellano. Acta, pp. 85–91.
16. **Haro, S. N. G., Gelbukh, A. (2007).** Investigaciones en análisis sintáctico para el

español. Instituto Politécnico Nacional, Dirección de Publicaciones.

18. **Edmundson, H. P. (1969).** New methods in automatic extracting. Journal of the ACM (JACM), Vol. 16, No. 2, pp. 264–285. DOI: 10.1145/321510.32151.
19. **Rush, J. E., Salvador, R., Zamora, A. (1971).** Automatic abstracting and indexing. II. Production of indicative abstracts by application of contextual inference and syntactic coherence criteria. Journal of the American Society for Information Science, Vol. 22, No. 4, pp. 260–274. DOI: 10.1002/asi.4630220405.
20. **Pollock, J. J., Zamora, A. (1975).** Automatic abstracting research at chemical abstracts service. Journal of Chemical Information and Computer Sciences, Vol. 15, No. 4, pp. 226– 232.
21. **DeJong, G. (1982).** An overview of the FRUMP system. Strategies for natural language processing, Vol. 113, pp. 149–176.
22. **Jones, K. S. (1993).** What might be in a summary? Information retrieval, Vol. 93, pp. 9– 26.
23. **Kupiec, J., Pedersen, J., Chen, F. (1995).** A trainable document summarizer. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval pp. 68–73.
24. **Endres-Niggemeyer, B. (1998).** Summarizing Information (Springer).
25. **Mani, I., Maybury, M. T. (1999).** Advances in automatic text summarization MIT Press. Vol. 293.
26. **Moens, M. F. (2000).** Automatic indexing and abstracting of document texts. Boston: Kluwer Academic Publishers.
27. **Mani, I. (2001).** Automatic summarization. John Benjamins. Retrieved from.
28. **Mani, I., House, D., Klein, G., Hirschman, L., Firmin, T., Sundheim, B. (1999).** The TIPSTER SUMMAC text summarization evaluation. Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics, pp. 77–85.
29. **Alfonseca, E., Rodríguez, P. (2003).** Generating extracts with genetic algorithms. Presented at the European Conference on Information Retrieval, Springer. pp. 511–519.
30. **Giannakopoulos, G., El-Haj, M., Favre, B., Litvak, M., Steinberger, J., Varma, V. (2011).** TAC 2011 MultiLing pilot overview.
31. **Elhadad, M., Miranda-Jiménez, S., Steinberger, J., Giannakopoulos, G. (2013).** Multi-document multilingual summarization corpus preparation, part 2: Czech, hebrew and spanish. Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization, pp. 13–19.
32. **Giannakopoulos, G., Kubina, J., Conroy, J., Steinberger, J., Favre, B., Kabadjov, M., Poesio, M. (2015).** Multiling 2015: multilingual summarization of single and multi-documents, on-line fora, and call-center conversations. Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 270–274.
33. **Over, P., Yen, J. (2001).** Introduction to DUC-2001: An intrinsic evaluation of generic news text summarization systems. Proceedings of DUC 2001 Document Understanding Conference, Vol. 49.
34. **Over, P., Liggett, W. (2002).** Introduction to DUC-2002: An intrinsic evaluation of generic news text summarization system. ACL 2002, Workshop on Text Summarization.
35. **Over, P., Yen, J. (2003).** Introduction to DUC-2003: An intrinsic evaluation of generic news text summarization systems.
36. **Over, P., Yen, J. (2004).** Introduction to DUC-2004: An intrinsic evaluation of generic news text summarization systems.
37. **Dang, H. T. (2005).** Overview of DUC 2005. Proceedings of the document understanding conference, Vol. 2005, pp. 1–12.
40. **Dang, H. T., Owczarzak, K. (2008).** Overview of the tac 2008 opinion question answering and summarization tasks. Proceedings of the First Text Analysis Conference, Vol. 2.
41. **Dang, H. T., Owczarzak, K. (2009).** Overview of the TAC 2009 summarization track. Proceedings of the Text Analysis Conference. pp. 1–25.

42. **Owczarzak, K., Dang, H. T. (2010).** Overview of the tac 2010 summarization track. Proceedings of the Third Text Analysis Conference, Gaithersburg, Maryland, USA, National Institute of Standards and Technology.
43. **Owczarzak, K., Dang, H. T. (2011).** Overview of the tac 2011 summarization track: Guide task and aesop task. Proceedings of the Second Text Analysis Conference (TAC2011). Gaithersburg, Maryland, USA.
44. **Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Stolcke, A. (2003).** The ICSI meeting corpus. 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 1, p. I–I.
45. **McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Karaiskos, V. (2005).** The AMI meeting corpus. Proceedings of the 5th international conference on methods and techniques in behavioral research, Vol. 88, p. 100.
46. **Ganesan, K., Zhai, C., Han, J. (2010).** Opinosis: A graph-based approach to abstractive summarization of highly redundant opinions. Proceedings of the 23rd international conference on computational linguistics, pp. 340–348.
47. **David, G., Cieri, C. (2003).** English gigaword. Linguistic Data Consortium.
48. **Parker, R., Graff, D., Kong, J., Chen, K., Maeda, K. (2011).** English gigaword fifth edition. Technical Report, Linguistic Data Consortium, Philadelphia.
49. **Hu, B., Chen, Q., Zhu, F. (2016).** LCSTS: A large scale chinese short text summarization dataset. Computation and Languaje, arXiv preprint arXiv:1506.05865.
50. **Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P. (2015).** Teaching machines to read and comprehend. Advances in neural information processing systems, Vol. 28, pp. 1693–1701.
51. **Toutanova, K., Brockett, C., Tran, K. M., Amershi, S. (2016).** A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs.
52. **Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., Goharian, N. (2018).** A discourse-aware attention model for abstractive summarization of long documents. arXiv preprint arXiv:1804.05685.
53. **El-Haj, M., Kruschwitz, U., Fox, C. (2010).** Using mechanical turk to create a corpus of arabic summaries.
54. **Radev, D. (2003).** Summbank 1.0. web download. Linguistic Data Consortium, Philadelphia.
55. **Hasler, L., Orasan, C., Mitkov, R. (2003).** Building better corpora for summarization. Proceedings of corpus linguistics. pp. 309– 319.
56. **Lins, R. D., Oliveira, H., Cabral, L., Batista, J., Tenorio, B., Ferreira, R., Simske, S. J. (2019).** The CNN-corpus: A large textual corpus for single-document extractive summarization. Proceedings of the ACM Symposium on Document Engineering. pp. 1- 10.
57. **Pardo, T. A. S., Rino, L. H. M. (2003).** TeMário: Um corpus para sumarização automática de textos. São Carlos: Universidade de São Carlos, Relatório Técnico.
58. **Acero, I., Alcojor, M., Díaz-Esteban, A., Gómez-Hidalgo, J. M., Maña-López, M. J. (2001).** Generación automática de resúmenes personalizados. Procesamiento del lenguaje natural, No 27, pp. 281–290.
59. **Toledo-Báez, M. C. (2010).** Aproximación al resumen automático como herramienta de ayuda a la traducción jurídica en el ámbito del Derecho turístico1. El español, lenguaje de traducción para la cooperación y el dialogo, Actas del IV Congreso El español, lengua de traducción, Madrid.
60. **Da-Cunha, I. (2008).** Hacia un modelo lingüístico de resumen automático de artículos médicos en español. Proyecto de investigación, Universidad Pompeu Fabra, Instituto Universitario de Lingüística Aplicada, Doctorado en Ciencias del Lenguaje y Lingüística Aplicada, http://www.tesisenxarxa.net.

**61. Villatoro, E. (2007).** Generación automática de resúmenes de múltiples documentos. Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla.

**62. Plaza, L. (2011).** Uso de grafos semánticos en la generación automática de resúmenes y estudio de su aplicación en distintos dominios: biomedicina, periodismo y turismo. Universidad Complutense de Madrid, Madrid.

**63. Venegas, R. (2011).** Evaluación de resúmenes en español con análisis semántico latente: Una implementación posible. Revista signos, Vol. 44, No. 75, pp. 85–102.

**64. Cabral, L. S., Lins, R. D., Mello, R. F., Freitas, F., Ávila, B., Simske, S., Riss, M. (2014).** A platform for language independent summarization. Proceedings of the 2014 ACM symposium on Document engineering, pp. 203–206.

**65. Lins, R. D., Oliveira, H., Cabral, L., Batista, J., Tenorio, B., Salcedo, D. A., Simske, S. J. (2019).** The CNN-corpus in spanish: a large corpus for extractive text summarization in the spanish language. Proceedings of the ACM Sym-posium on Document Engineering, pp. 1– 4.

**66. Matias, G. A., Ledeneva, Y., García-Hernández, R. A., Alexandrov, M., Hernández-Castañeda, Á. (2020).** Ground Truth Spanish Automatic Extractive Text Summarization Bounds. Computación y Sistemas, Vol. 24, No. 3, pp. 1241–1256. DOI: 10.13053/CyS-24-3-3484.

**67. Scialom, T., Dray, P. A., Lamprier, S., Piwowarski, B., Staiano, J. (2020).** MLSUM: The multilingual summarization corpus. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing EMNLP, pp. 8051–8067.

**68. Segarra-Soriano, E., Ahuir, V., Hurtado, L. F., González, J. (2022).** DACSA: A large-scale dataset for automatic summarization of catalan and spanish newspaper articles. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5931– 5943.

**69. Bernoldi, R., Tolosa, G. H. (2022).** Generación automática de resúmenes abstractivos de noticias en español. Presented at the Simposio Argentino de Ciencia de Datos y Grades Datos.

**70. Sparck-Jones, K., Galliers, J. R. (1995).** Evaluating natural language processing systems: An analysis and review Springer Science & Business Media, Vol. 1083.

**71. Berker, M., Güngör, T. (2012).** Using genetic algorithms with lexical chains for automatic text summarization. ICAART, Vol. 1, pp. 595–600.

**72. Lin, C. Y., Hovy, E. (2003).** Automatic evaluation of summaries using n-gram co-occurrence statistics. Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Vol. 1, pp. 71–78.

**73. Lin, C. Y., Och, F. J. (2004).** Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics. p. 605.

**74. Téllez, A., Montes, M., Villaseñor-Pineda, L. (2009).** Using machine learning for extracting information from natural disaster news reports. Computación y Sistemas, Vol. 13, No. 1, pp. 33–44.

**75. Da-Cunha, I., Torres-Moreno, J. M., Velázquez-Morales, P., Vivaldi, J. (2009).** Un algoritmo lingüístico-estadístico para resumen automático de textos especializados. Linguamática, Vol. 1, No. 2, pp. 67–79.

**76. Molina, A. (2013).** Compresión automática de frases: un estudio hacia la generación de resúmenes en español. Inteligencia Artificial, Vol. 16, No. 51, pp. 41–62.

**77. Sekine, S., Nobata, C. (2003).** A survey for multi-document summarization. Proceedings of the HLT-NAACL 03 on Text summarization workshop, Vol. 5, pp. 65–72.

**78. Das, D., Martins, A. F. (2007).** A survey on automatic text summarization. Literature

Survey for the Language and Statistics II course at CMU, Vol. 4, pp. 192–195.

79. **Gholamrezazadeh, S., Salehi, M. A., Gholamzadeh, B. (2009).** A comprehensive survey on text summarization systems. Computer Science and its Applications, 2009. CSA'09. 2nd International Conference on pp. 1–6.
80. **Gupta, V., Lehal, G. S. (2010).** A survey of text summarization extractive techniques. Journal of emerging technologies in web intelligence, Vol. 2, No. 3, pp. 258–268.
81. **Damova, M., Koychev, I. (2010).** Query-based summarization: A survey.
82. **Nenkova, A., McKeown, K. (2012).** A survey of text summarization techniques. Mining text data, Springer, pp. 43–76.
83. **Elfayoumy, S., Thoppil, J. (2014).** A survey of unstructured text summarization techniques. International Journal of Advanced Computer Science and Applications, Vol. 5, No. 4.
84. **Saranyamol, C. S., Sindhu, L. (2014).** A survey on automatic text summarization. International Journal of Computer Science and Information Technologies, Vol. 5, No. 6, pp. 7889–7893.
85. **Al-Saleh, A. B., Menai, M. E. B. (2016).** Automatic arabic text summarization: a survey. Artificial Intelligence Review, Vol. 45, No. 2, pp. 203–234.
86. **Gambhir, M., Gupta, V. (2017).** Recent automatic text summarization techniques: a survey. Artificial Intelligence Review, Vol. 47, No. 1, pp. 1–66.
87. **Al-Qassem, L. M., Wang, D., Al-Mahmoud, Z., Barada, H., Al-Rubaie, A., Almoosa, N. I. (2017).** Automatic arabic summarization: a survey of methodologies and systems. Procedia Computer Science, Vol. 117, pp. 10– 18.
88. **Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., Kochut, K. (2017).** Text summarization techniques: a brief survey. arXiv pre-print arXiv:1707.02268.
89. **Téllez-Valero, A., Montes-y-Gómez, M., Villaseñor-Pineda, L. (2009).** Using machine learning for extracting information from natural disaster news reports. Computación y sistemas, Vol. 13, No. 1, pp. 33–44.
90. **Cardoso, A. C., Abelleira, M. A. P. (2013).** Generación automática de resúmenes. 1er Congreso Nacional de Ingeniería Informática / Sistemas de Información, CoNaIISI.
91. **Careaga-Moya, J. A., Medina-Urrea, A., Torres-Moreno, J. M. (2012).** A new cross-lingua automatic summarization approach based on textual energy. Journées internationales d'Analyse statistique des Données Textuelles, pp. 247–255.
92. **Fernández, S., SanJuan, E., Torres-Moreno, J. M. (2007).** Textual energy of associative memories: Performant applications of Enertex algorithm in text summarization and topic segmentation. Mexican International Conference on Artificial Intelligence, Springer, pp. 861–871.
93. **Leiva-Mederos, A., Domínguez-Velasco, S., Senso, J. A. (2012).** PuertoTex: a data mining software based on ontologies for automatic summarization on port and coastal engineering domain. Transinformação, Vol. 24, No. 2, pp. 103–115.
94. **Plaza-Morales, L. (2008).** Generación automática de resúmenes con apoyo en ontologías aplicada al dominio biomédico.
95. **Umadevi, K. S., Chopra, R., Singh, N., Aruru, L., Kannan, R. J. (2018).** Text summarization of Spanish documents. 2018 International Conference on Advances in Computing, Communications and Informatics pp. 1793–1797.
96. **Caparrós-Laiz, C., García-Díaz, J. A., Valencia-García, R. (2021).** Evaluating extractive automatic text summarization techniques in spanish. Technologies and Innovation, pp. 79–92.
97. **Cañete, J., Chaperon, G., Fuentes, R., Ho, J. H., Kang, H., Pérez, J. (2023).** Spanish pre-trained BERT model and evaluation data. DOI: 10.48550/arXiv.2308.02976.
98. **Vogel-Fernandez, A., Calleja, P., Rico, M. (2022).** esT5s: A spanish model for text summarization. Towards a Knowledge-Aware AI (pp. 184–190).

**99. Hasan, T., Bhattacharjee, A., Islam, M. S., Samin, K., Li, Y. F., Kang, Y. B., Shahriyar, R. (2021).** XL-Sum: Large-scale multilingual abstractive summarization for 44 languages.

**100. Mihalcea, R., Tarau, P. (2005).** A language independent algorithm for single and multiple document summarization.

**101. Patel, A., Siddiqui, T., Tiwary, U. S. (2007).** A language independent approach to multilingual text summarization. Presented at the Large-scale semantic access to content (text, image, video, and sound), pp. 123–132.

**102. Last, M., Litvak, M. (2010).** Language-independent techniques for auto-mated text summarization. pp. 207–237.

**103. Saggion, H. (2011).** Using SUMMA for language independent summarization at TAC 2011. Presented at the TAC.

**104. El-Haj, M., Rayson, P. (2013).** Using a keyness metric for single and multi-document summarization. Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization, pp. 64–71.

*Article received on 27/07/2022; accepted on 13/09/2024.*
**Corresponding author is Yulia Ledeneva.*

ISSN 2007-9737

# Unsupervised Keyphrase Extraction: Ranking Step and Single-Word Phrase Problem

Svetlana Popova[1,*], Vera Danilova[2], Mikhail Alexandrov[3,4], John Cardiff[1]

[1] Technical University of Dublin, Ireland

[2] Uppsala University, Sweden

[3] Russian Academy of National Economy and Public Administration, Moscow, Russia

[4] FRUCT Association, Helsinki, Finland

spbu.svp@gmail.com, vera.danilova@lingfil.uu.se, malexandrov@mail.ru, john.cardiff@tudublin.ie

**Abstract.** Keyphrases provide a compact representation of a document's content and can be efficiently used to enhance Web search results and improve natural language processing tasks. This paper extends the state-of-the-art in unsupervised keyphrase extraction from scientific abstracts. We aim to demonstrate the difference between two types of datasets used in the keyphrase extraction domain: datasets where keyphrases for each text are manually assigned by readers, and datasets where keyphrases are assigned by the authors themselves. We aim to highlight the problem of single-word phrases and illustrate the role of this problem for each dataset type. Additionally, we noticed that well-known algorithms in the domain can be divided into two groups. Algorithms in the first group minimize the number of single-word phrases in the set of the extracted keyphrases. In contrast, algorithms in the second group allow the extraction of a larger number of single-word keyphrases. This property of algorithms "to extract few or many single-word keyphrases" determines how they perform on each type of dataset. We explain the reasons for this.

**Keywords.** Unsupervised keyphrase extraction, single-word phrase problem, keyphrase length.

## 1 Introduction

Keyphrases provide a compact representation of a document's content [32] and automatic keyphrase extraction (**KE**) concerns "the automatic selection of important and topical phrases from the body of a document" [36]. We distinguish KE from keyphrase assignment and keyphrase generation:

1. Assignment: Keyphrases are assigned to a text from a predefined controlled vocabulary.
2. Generation: The extracted phrases do not necessarily appear in a given document during keyphrase generation.

In KE, the extracted keyphrases should occur in the document and are not restricted by a predefined vocabulary. In Table 1 keyphrases are emphasized in bold for the following example abstract. Good keyphrases should satisfy the following properties [26]: Meaningfulness, relevance, and good coverage.

Being more descriptive than single keywords, keyphrases help to perform efficient text mining and are involved in improving the functionality

**Table 1.** Scientific abstracts and keyphrases (INSPEC dataset)

| |
|---|
| **Accelerated simulation** of the steady-state availability of non-Markovian systems. A **general accelerated simulation method** for evaluation of the **steady-state availability** of **non-Markovian systems** is proposed. It is applied to the investigation of a class of systems with repair. **Numerical examples** are given. |

of information retrieval systems [17, 3, 39]. KE methods are divided into supervised and unsupervised. Unsupervised methods can be straightforwardly applied to documents and do not require training data.

Our research entirely focuses on unsupervised approaches. Also, we constrain the domain to titles and abstracts of scientific publications. KE from this type of text has been the focus of researchers for a long time due to the development of electronic libraries and e-learning systems. KE methods can be divided into word-based and candidate-based.

– **Word-based** methods consist of two stages: (1) document terms are weighed and then ranked to select words that belong to keyphrases and, (2) keyphrases are built from selected terms. Often it means merging terms that follow each other in a given text.
– **Candidate-based** methods include two stages: (1) keyphrase candidates extraction, and (2) keyphrase selection based on the ranking of the candidates.

Most unsupervised KE algorithms are candidate-based. We only explore candidate-based approaches in this research.

KE methods have traditionally been evaluated and compared to other methods based on the F1-score [21], which has become the standard and consistently used evaluation approach in the KE domain. Whenever we refer to the performance of the KE algorithm, its quality, or its evaluation score, we mean the evaluation of its performance using the macro-average F1-score. This score will be discussed below.

While researchers have dedicated efforts to enhance the final KE evaluation scores produced by previous algorithms, there appears to be a gap in addressing the underlying causes for variations in ranking quality observed among these algorithms. To the best of our knowledge, the factors contributing to these differences in ranking performance have not been deeply explored yet. This study partially deals with this problem. In addition, we generalize the properties of different datasets and categorize them into two groups with distinct characteristics.

This work brings the following new contributions:

1. We show that the advantage or weakness of an algorithm over the others can be determined by the number of single-word phrases that this algorithm extracts as keyphrases.
2. We highlight the difference between two types of datasets used in the KE domain: datasets where keyphrases for each text are manually assigned by readers, and datasets where keyphrases are assigned by the authors themselves. We demonstrate that datasets with reader-assigned keyphrases tend to exhibit a problem related to single-word phrases, the extent of which varies by dataset. Specifically, algorithms that extract fewer single-word phrases often perform better in evaluations, as accurately extracting single-word keyphrases from a set of candidates is more difficult than it is for multi-word phrases.
3. In this work, we demonstrate that it is possible to improve the quality of the existing algorithms by considering the properties of single-word phrases, especially for datasets where the single-word phrase problem is strongly pronounced.

## 2 Related Work

This section will focus on

1. Unsupervised KE algorithms.
2. On the length feature in KE algorithms.

We divide unsupervised KE methods into graph-based, statistics-based, and embedding—transformer-based methods. Then we will highlight the length feature that is actively used. In the review, we primarily focus on the algorithms used in the study. These algorithms are described in more detail than the others.

### 2.1 Graph-based Unsupervised Methods

The original TextRank [28] is word-based and this algorithm can be defined as a baseline for graph-based methods. Firstly, it creates a graph to rank words, where words are vertices and edges represent the fact of word co-occurrence in a given text within a window of size $n$ (edges are not weighted). The words (vertices of the graph) are weighted and ranked.

To calculate the word weight, a modification of the PageRank formula [8] is used. Then one-third of the top-ranked words are selected, which are denoted as one-word keyphrases or merged into multi-word keyphrases if they follow each other in the text. The authors report that the best results are achieved with $n = 2$ when only nouns and adjectives are allowed in the keyword set [28].

The approach proposed for graph construction and the vertices weighting is actively used in the KE domain. While the vertices can be text words, candidate phrases, noun phrases, or document topics, the edges may or may not be weighted and represent various types of connections between nodes, such as the co-occurrence of words or semantic similarity of units in the nodes.

SingleRank [37] is a candidate-based approach. Keyphrase candidates are extracted as the longest continuous sequences of nouns and adjectives in a given text. Phrases ending with an adjective are not allowed. The score of candidates is calculated by summing the scores of the words it contains. Word scores are counted recursively, similar to PageRank [8] modification in TextRank [28], using a local graph for a given document that includes only nouns and adjectives as vertices and weighs edges based on word co-occurrence. ExpandRank presented in the same article [37] is similar but it uses $k$ most similar documents to weigh the edges based on word co-occurrence.

In PositionRank [16], phrases are built as contiguous noun phrases that match the pattern (adjective)*(noun)+ and have a length of up to three words. The weight of a phrase is calculated as the sum of the weights of its constituent words. The weight of each word in a phrase is calculated by building a word graph and using a modified version of PageRank, similar to TextRank [28], but taking into account the position of each word in the text. The idea behind the algorithm is to assign higher weights to words that are frequent and appear early in documents.

Another part of the graph-based algorithms uses a modification of the PageRank formula and integrates topic information in it in various ways. In [25], topics are first defined based on LDA [4]. The words are then weighed using a modified formula for each of the topics. That is, the word will have different weights regarding different topics. These scores are used to weigh the phrases. Candidate phrases in this method are extracted as noun phrases. A candidate's weight is calculated as the sum of the scores of the words included in the phrase. Since the score of a word varies across different topics, the weight of the candidate will be different relative to different topics. Candidate weights across different topics are combined into a final phrase weight by considering the document's topic distribution. All candidate phrases are ranked based on their final weights, and top-ranked phrases are selected as keyphrases.

In TopicRank [7] candidate phrases for a given document are defined as the longest sequence of nouns and adjectives. The algorithm extracts candidates and groups them by topic using hierarchical agglomerative clustering. A graph-based ranking model based on the PageRank formula [8] is applied to assign a weighted score to each topic. Here, topics are vertices, and edges are weighted according to the strength of semantic relations between topics, which depends on how often these topics' keyphrase candidates appear close to each other in the document. Keyphrases are then extracted by selecting a candidate from each of the top-ranked topics. Three strategies were considered for selecting a candidate for each topic. The first is to

choose the candidate that appears earliest in the document. The second is to choose the candidate with the highest frequency of occurrence. The third is to select the centroid of the topic cluster. The authors showed that selecting the first occurrence of a keyphrase candidate is the best strategy of the three.

In MultipartiteRank [6], candidate phrases are built as continuous noun phrases that match the pattern (adjective)*(noun)+. Candidates are grouped into topics with hierarchical agglomerative average linkage clustering. The graph is constructed as follows: candidate phrases are vertices that are connected if they belong to different topics. The authors adjust the incoming edge weights of the nodes that correspond to the first occurrence for each topic. Thus, candidates that occur at the beginning of the text are promoted in comparison with other candidates from the topic. When the graph is built, keyphrase candidates that are nodes of the graph are ranked based on a modification of the PageRank algorithm. Top-ranked candidates are selected as keyphrases.

### 2.2 Statistical and Embedding/Transformer Based Unsupervised Methods

RAKE [32] extracts candidate keyphrases from a text by identifying the longest sequences of continuous words that are split at phrase delimiters, stop-word positions, and word delimiters. RAKE proposed a method to extract additional words that act as delimiters between phrases. The weight of a candidate is determined by summing the weights of its words. To estimate the weights, it uses a graph of word co-occurrences within extracted candidates. Word's score is calculated based on the ratio of its degree in the obtained graph to its frequency in the text. Since the degree of a graph vertex is the number of graph edges incident to this vertex, then the degree of a vertex word shows the number of words with which this word co-occurs within the extracted candidates. So, the total score of a word is computed by dividing the number of co-occurring words with that particular word by its frequency. That is why we can exclude graph-based notation here and look at this approach as statistical. In contrast, in previously described graph-based methods we could not exclude graph-based notation.

KP-miner [15] extracts candidate phrases in two main steps. First, it identifies n-grams by extracting sequences of words separated by punctuation marks and stop words. It then applies two additional filters to these candidates. The first filter requires each phrase to occur at least $n$ times in the text, where $n$ depends on the length of the document. The second filter considers the position of the phrase within the text. It is controlled by the CutOff parameter. To filter the phrases, it uses the number of words encountered before the first occurrence of a phrase. In the second step, the algorithm scans the text again and selects, from the unfiltered units, the sequences of maximum length as candidates.

KP-miner uses Tf-Idf for ranking. Since the problem with this method is that it prefers single-word phrases, which are the most frequently generated, the authors introduce a special boosting factor for multi-word phrases. It allows the algorithm to balance this bias towards single terms. In addition, the document frequency of multi-word phrases is assumed to be equal to 1 when calculating their Idf. This gives an advantage to multi-word phrases over single-word ones.

YAKE [9] uses a sliding window of 3-g generating a contiguous sequence of 1, 2, and 3-g candidate keywords. Candidates that begin or end with a stop word are not allowed. YAKE exploits the following features: casing, word position, word frequency, word relatedness to context, and word difference. These features are combined within a single complex formula [10, 11].

The following algorithms involve word embeddings and transformer models into KE. Candidates are commonly extracted as noun phrases, sequences of nouns and adjectives. EmbedRank [2] is an embedding-based method. It allows for the calculation of the distance between the embedding of a candidate and that of the entire document, as well as between candidates themselves. To achieve this, the authors represent the candidates and the document in the same high-dimensional vector space and utilize publicly available pre-trained models of Sent2Vec [29]

and Doc2Vec [22]. Keyphrases were selected by ranking the candidate phrases according to their cosine distance to the document embedding.

AttentionRank [13] is based on the self-attention mechanism of the BERT model [12], as well as the hierarchical attention retrieval (HAR) mechanism [40]. Self-attention determines the importance of a candidate within the context of a sentence, while cross-attention measures the semantic relevance between a candidate and sentences within a document. These two values are used to calculate the final score of a candidate.

In the TripleRank [23] method two features: keyphrase semantic diversity and keyphrase coverage are introduced to address the issue of synonyms, along with positional information. Keyphrase coverage calculates the similarity between candidates and other words in the document using Word2Vec, while semantic diversity uses LDA [4] to diversify the topics represented in phrases and avoid extracting phrases from the same words.

The sentence embedding model SIF [35] from SIFRank is intended to explain the relationship between sentence embeddings and the topic of the document. The authors combine ELMo [30] with SIF to compute phrase embeddings and document embeddings. To weigh the phrases, cosine similarity is used to calculate the distance between the candidates and the topic. The paper also proposes a method called document segmentation to speed up the computation of word embeddings in long documents and uses position-biased weight for long documents. Further development of embedding/ transformer-based KE algorithms are presented in the [24, 14, 34].

### 2.3 Phrase Length Feature

One of the employed techniques in candidate weighting is the method, which assigns weights to candidates by summing the scores of the constituent words. This is exploited in SingleRank [37], PositionRank, Topical PageRank [16, 25], RAKE [32] and in the re-implementation of TextRank that is exploited in this paper (will be discussed below). Since all words have positive scores, we expect these algorithms to prioritize multi-word phrases over single-word phrases in the weighting process. As we will see later, the implementations of these algorithms we used either do not extract single-word keyphrases at all or extract only a minimal number of them. This will differentiate them and their performance from other algorithms that extract a greater number of single-word phrases.

KP-miner [15] also employs features that limit the extraction of single-word phrases, but as we will see later, despite these features, when processing abstracts of scientific publications, KP-miner extracts more single-word phrases than the methods discussed in the previous paragraph. This algorithm introduces a special boosting factor for multi-word phrases. Besides, KP-miner exploits Tf-Idf for candidate ranking where the document frequency of multi-word phrases is assumed to be equal to 1 when calculating their Idf. Both these factors give an advantage to multi-word phrases over single-word ones.

In the YAKE algorithm, there is also a remark [9] regarding the formula used to calculate the weight of a candidate phrase, which combines the features employed in the algorithm. In particular, one part of the formula is considered and explained: one of the potential problems with this part of the formula is that the length of a candidate keyphrase can significantly influence its score, favoring longer relevant candidates over shorter relevant ones. To mitigate this effect and extract keyphrases regardless of their length, YAKE introduces additional components into the general formula. However, as we will see later with short texts, YAKE significantly limits the number of extracted single-word phrases. It extracts more of them than the approaches from the first paragraph but fewer than, for example, KP-Miner.

This section aims to demonstrate that algorithms incorporate features and methods influencing the number of single-word keyphrases they extract. Additionally, we hypothesize that algorithms can be grouped based on their tendency to extract more or fewer single-word keyphrases. The central focus of this article is to examine how the tendency of algorithms to select

more or fewer single-word phrases impacts the evaluation of their performance.

## 3 Research Tools

As mentioned in the Introduction, this article focuses solely on candidate-based KE approaches. This means that each exploited below algorithm operates as follows. In the first step, a set of candidate phrases is constructed. In the second step, this set of candidate phrases is ranked, and the top k-ranked phrases are extracted as keyphrases.

For all the algorithms in the experiments, we use the same method to extract the set of candidate phrases. The candidate phrases are extracted as noun phrases using grammar, the description of which is provided below. This method was chosen because it allows us to obtain a good quality set of candidate phrases compared to using other methods, such as extracting n-grams or extracting sequences of words between delimiters. By ”good quality” we mean that the set of candidate phrases should not be too large on the one hand, and on the other hand, it should contain a sufficient number of true positive keyphrases. In the KE domain, it has been noted that the size and quality of the set of candidate phrases affect the quality of the keyphrases obtained after ranking [38]. By quality here, we mean the evaluation with F1-score. Additionally, the vast majority of approaches reviewed in the Related Work extract candidate phrases as noun phrases or as sequences of nouns and adjectives that do not end with an adjective.

### 3.1 PKE, Algorithms and Modifications, Text Pre-Processing

PKE [5] is an end-to-end Python KE pipeline that includes the re-implementation of KE algorithms. PKE allows using original versions of algorithms and modifications of its components. Our goal was to evaluate how well the ranking methods performed in KE approaches. To check this, every ranking method should take the same set of candidate phrases as input.

We use a modified version of TextRank. TextRank is a word-based approach but PKE allows us to work with it as with a candidate-based method [5]. TextRank re-implementation in the PKE framework is as follows: words are ranked and then keyphrase candidates are either composed from the T-percent highest-ranked words as in the original paper or extracted using the candidate selection method. In the latter, candidates are ranked using the sum of their words' weights. We exploit the second opportunity.

There are candidate-based methods that extract keyphrase candidates either as n-grams or as combinations of n-grams, filters, and sequences, including YAKE, TF-IDF, and KP-miner. To evaluate the ranking strategies, the candidate extraction method for these approaches was kept consistent with that used for the other methods: candidates were extracted as noun phrases.

We exploit the following methods implemented in PKE: graph-based (TextRank, SingleRank, TopicRank, PositionRank, MultipartiteRank, TopicalPageRank) and statistics-based (FirstPhrase, TfIdf, KP-miner, YAKE). The FirstPhrases method extracts a fixed number of phrases that are met at the beginning of a given document. The TfIdf method assigns a weight to each candidate phrase using the TfIdf formula and then ranks the phrases based on these weights. The ranking strategies of other exploited KE algorithms are described in the Related Work.

In all experiments, candidate phrases are extracted from texts as continuous sequences of nouns and adjectives satisfying the default PKE grammar:

```
[NBAR: {<NOUN|PROPN|ADJ>*<NOUN|PROPN>},
NP: {<NBAR>}{<NBAR><ADP><NBAR>}]
```

where nouns and adjectives are not stop words. We exploited the extended stop word list from [31], which was applied to all algorithms and across all datasets. Consequently, all exploited methods differ only in the ranking step.

### 3.2 Datasets

The initial hypothesis of this study was that datasets with keyphrases annotated by authors differ from those annotated by readers. Therefore, we initially categorized the datasets into two groups: datasets with reader-assigned keyphrases and datasets with author-assigned keyphrases. The first group includes INSPEC, SemEval 2010 (only reader-assigned keyphrases), and SemEval 2017. The second dataset group comprises kp20k, PubMed, and KPBiomed.

**Datasets with reader-assigned keyphrases**. From *INSPEC* [1] [19] we use a test subset comprising 500 scientific publication titles with abstracts, each with reference keyphrases manually assigned by readers. Dataset domains: Computers and Control and Information Technology. The average number of reference keyphrases per text in the INSPEC dataset is $9.5$.

*SemEval2010* [2] [20] test subset containing 100 full-text documents. Only reader-assigned keyphrases were used as reference keyphrases. We utilized the SemEval 2010 dataset in the following format: SemEval 2010 (TA) includes only the titles and abstracts of articles. Dataset domains: Distributed Systems, Information Search and Retrieval, Distributed Artificial Intelligence - Multiagent Systems, Social and Behavioral Sciences - Economics. The average number of reference reader-assigned keyphrases per text in the SemEval2010 dataset is $12.4$.

*SemEval2017* [1]. A corpus for this task was constructed from open-access publications available on ScienceDirect. Each document consists of one paragraph of text drawn from a scientific paper. A total of 500 paragraphs from journal articles, distributed among the domains of Computer Science, Material Sciences, and Physics, were selected. We utilized the dataset available in the YAKE-associated repository [3]. The first 200 documents were designated as the test dataset. The average number of reference keyphrases per text in the SemEval 2017 dataset is $11.5$.

**Datasets with author-assigned keyphrases**. We exploited the first 2,000 documents from the test subsets of *kp20k* [4] [27]. Each document comprises a title and abstract from scientific articles and includes author-assigned keyphrases. Dataset domain: Computer Science. The average number of reference keyphrases per text in the kp20k dataset is $5.3$.

Dataset *PubMed* [5] [33] contains 1,320 articles with full text and author-assigned keyphrases. Titles are separated from full texts but abstracts are not. For each text, we took the title and the first 1,200 characters of the full text, assuming that in this way we would be able to use most of the abstracts. We exploited the first 500 documents from the database as a test collection. Dataset domain: Biomedical. The average number of reference keyphrases per text in the PubMed dataset is $5.4$.

*KPBiomed*[6] [18]: the first 2,000 documents from the test subset of this dataset were used. Each document includes a title and abstract from a scientific article and author-assigned keyphrases. Dataset domain: Biomedical. The average number of reference keyphrases per text in the PubMed is $5.3$.

### 3.3 Evaluation

Since this study is based on PKE, we utilize its built-in evaluation method to calculate the macro-average F1-score at $n$ keyphrases (F@n). The F1-score is the most standard and widely used evaluation approach in the KE domain. The value of $n$ indicates that the KE algorithm should select up to $n$ phrases based on the ranking results, or fewer if selecting $n$ phrases is not feasible. This evaluation scores compares the set of automatically extracted keyphrases with the set of reference keyphrases annotated by experts, readers, or authors.

We use $n = 3$ and $n = 5$ for datasets with author-assigned keyphrases, as the average

[1] https://huggingface.co/datasets/taln-ls2n/inspec
[2] https://huggingface.co/datasets/taln-ls2n/semeval-2010-pre
[3] https://github.com/LIAAD/KeywordExtractor-Datasets/blob/master/datasets/SemEval2017.zip
[4] https://huggingface.co/datasets/taln-ls2n/kp20k
[5] https://huggingface.co/datasets/taln-ls2n/pubmed
[6] https://huggingface.co/datasets/taln-ls2n/kpbiomed

number of keyphrases in the references for these collections is approximately five. For datasets with reader-assigned keyphrases, we use $n = 5$ and $n = 10$, as the average number of keyphrases in their references is approximately ten to twelve

Extracted phrases and reference phrases are stemmed before evaluation. We do not remove phrases that do not occur in the corresponding text from reference keyphrases. This practice exists in the domain, e.g., in [19, 37] and it makes Recall and F1-score higher.

## 4 Experiment Description, Results, and Discussion

We aim to accomplish the following in this section:

- *Examining dataset characteristics*: we analyze the characteristics of the datasets that may influence the quality of algorithm performance.
- *Formulating the single-word phrase problem*: we briefly outline and formulate the problem of single-word phrases.
- *Unsupervised KE algorithms evaluation*: we evaluate how 10 selected unsupervised KE algorithms perform on each dataset group (datasets with reader-assigned and author-assigned keyphrases) and examine the role single-word phrases play in their performance within each dataset group.

The length of a phrase refers to the number of words it contains. Reference keyphrases are manually assigned to a text and represent the keyphrases that the KE algorithm should ideally extract. All KE algorithms exploited in this study operate in two steps. In the first step, a set of candidate phrases is extracted, a process identical across all considered algorithms. In the second step, each algorithm ranks the set of candidate phrases.

### 4.1 Dataset Characteristics

In this part of the study, we collected the following characteristics for each dataset:

- *Reference Keyphrases*: the average number of reference keyphrases of lengths 1, 2, and 3 per document in the dataset.
- *Candidate Phrases*: the average number of candidate phrases per document of lengths 1, 2, and 3, extracted as continuous sequences of nouns and adjectives satisfying the chosen grammar.
- *Evaluation*: the Precision and F1-scores obtained when evaluating the quality of candidate phrases of lengths 1, 2, and 3. The quality of single-word phrases in the candidate set was evaluated against the single-word phrases in the reference set. Similarly, the quality of two-word candidate phrases was evaluated by comparison with two-word phrases in the reference set. The same procedure was applied to phrases with length 3.

The obtained results are presented in the Table 2. Analysis of Table 2 reveals the following differences between the two groups of datasets in the distribution of keyphrases length 1 and 2 in references and candidates.

**In reference keyphrase set** we observe the following. *Datasets with author-assigned keyphrases*: in datasets where authors assigned keyphrases, the average number of single-word and two-word phrases per text in the references is roughly equal, with an average difference of $0.45$ keyphrases. In the case of kp20k, this difference is greater than in PubMed and KPBiomed.

*Datasets with reader-assigned keyphrases*: in datasets where readers assigned keyphrases, the difference in the number of single-word and two-word phrases in the references is more pronounced, with an average difference of $2.86$, indicating a higher prevalence of two-word phrases. However, for SemEval2017, this difference is somewhat smaller compared to INSPEC and SemEval2010.

**In keyphrase candidates set** we observe the following. The candidate phrase sets for

**Table 2.** Dataset Characteristics. Ref: the average number of reference keyphrases of a specified length per document. Cand: the average number of candidate phrases of a specified length per document. P: Precision. F1: F1-score. Reader keys: indicates datasets where readers assigned keyphrases. Author keys: indicates datasets where authors assigned keyphrases. SemEval2010: SemEval2010(TA) contains only titles and abstracts

| | INSPEC reader keys | | | | SemEval2010 reader keys | | | | SemEval2017 reader keys | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| len. | Ref | Cand | P | F1 | Ref | Cand | P | F1 | Ref | Cand | P | F1 |
| 1 | 1.31 | 11.85 | 8.64 | 13.43 | 1.99 | 16.18 | 4.87 | 8.02 | 3.95 | 16.23 | 17.04 | 25.04 |
| 2 | 5.16 | 8.58 | 40.64 | 46.99 | 5.54 | 11.38 | 16.09 | 19.70 | 5.13 | 11.765 | 31.98 | 40.55 |
| 3 | 2.44 | 2.82 | 40.56 | 40.18 | 1.98 | 3.49 | 12.06 | 12.57 | 2.78 | 4.175 | 34.43 | 35.82 |

| | PubMed author keys | | | | KPBiomed author keys | | | | kp20k author keys | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| len. | Ref | Cand | P | F1 | Ref | Cand | P | F1 | Ref | Cand | P | F1 |
| 1 | 3,93 | 18,7 | 6.30 | 10.56 | 2,18 | 22,75 | 5.16 | 8.79 | 1.75 | 14.08 | 5.19 | 7.12 |
| 2 | 3,34 | 12,66 | 6.42 | 10.46 | 2,11 | 16,58 | 6.01 | 9.91 | 2.44 | 10.86 | 9.13 | 13.17 |
| 3 | 1,15 | 4,34 | 4.77 | 6.54 | 0,71 | 5,5 | 4.89 | 6.90 | 0.78 | 3.87 | 7.02 | 8.86 |

all collections were extracted uniformly, and all datasets primarily consisted of titles and abstracts. Consequently, the proportion of single-word and two-word phrases in the candidate sets is similar across all datasets. In each dataset, the number of single-word phrases exceeds that of two-word phrases by a factor of $1.3$ to $1.5$. Despite the similar proportions of single-word and two-word phrases in the datasets, the evaluation of these sets using Precision and F1-score differs, which is partly a consequence of the differences in proportions between reference keyphrases of lengths 1 and 2.

*Datasets with author-assigned keyphrases*: the disparity between Precision and F1-score values for single-word and two-word phrases, when keyphrases are assigned by authors, is not as significant as in the case of reader-assigned keyphrases. Specifically, the difference in Precision is $0.12$ for PubMed, $0.85$ for KPBiomed, and $0.94$ for kp20k. In terms of the F1-score, the differences are $0.1$ for PubMed, $1.12$ for KPBiomed, and $6.05$ for kp20k.

*Datasets with reader-assigned keyphrases*: the disparity between Precision and F1-score values for single-word and two-word phrases, when keyphrases are assigned by readers, is very significant. Specifically, in terms of Precision, this difference is $32$ for INSPEC, $11.22$ for SemEval2010(TA), and $14.94$ for SemEval2017. In terms of F1-score: $33.56$ for INSPEC, $11.68$ for SemEval2010(TA), and $15.51$ for SemEval2017.

For datasets where readers assigned keyphrases, the number of keyphrases in the references is approximately twice as large as in the references of texts where authors assigned keyphrases. This difference is attributable to the greater number of multi-word phrases. Two-word phrases are less represented in the reference sets of author-annotated datasets than in reader-annotated datasets. Furthermore, for datasets where keyphrases were assigned by authors, Precision and F1-score for two-word candidate phrases are low: in two out of three cases, they are lower than the corresponding values for single-word phrases.

### 4.2 Single-Word Phrase Problem

The main points from the previous section are:

1. *Datasets with reader-assigned keyphrases*: in the reference sets of these datasets there are more multi-word phrases compared to single-word phrases.
2. In the candidate phrase sets, the number of single-word phrases is $1.3$ to $1.5$ times higher than that of two-word phrases.

3. *Datasets with reader-assigned keyphrases*: Precision and F1-score calculated for single-word candidate phrases relative to the set of single-word keyphrases in the references are significantly lower than those for two-word and three-word phrases.

The essence of the single-word phrase problem for datasets with reader-assigned keyphrases: the initial set of candidate phrases contains a significant number of single-word phrases, and for single-word phrases in the candidate set, the true positive percentage may be significantly lower than for multi-word phrases. This problem is dataset-dependent. Since some of the main datasets in the KE domain present this problem it is worth paying attention to. The consequences of the problem: it is more difficult to extract true positive single-word keyphrases than the same for multi-word phrases. Hypotheses based on this problem: KE algorithms can achieve higher performance not only by selecting well-suited linguistic features but also by minimizing the number of extracted single-word keyphrases.

### 4.3 Unsupervised KE Algorithms Evaluation and Single-Word Phrase Problem

To demonstrate the connection between the total number of single-word phrases in the resulting keyphrase set and the algorithm's performance, we conducted the following experiment. We ran 10 unsupervised KE algorithms on all six datasets and collected the following:

– The average number of single-word keyphrases extracted per text by the algorithm.

– The quality of the algorithm's performance, evaluated using the F1-score, without any restrictions on the algorithm's operation.

– The quality of the algorithm's performance, evaluated using the F1-score, when the algorithm is restricted to extracting only multi-word keyphrases.

The results are presented in the Table 3. In the l-1 phr. columns, which show the average number of single-word phrases extracted per text by the algorithm, five cases with the highest number of such phrases are highlighted in bold. Corresponding to these highlights, in the rows, instances where prohibiting the extraction of single-word phrases led to an improvement in the algorithm's performance are also highlighted in bold. The top five scores obtained for each specific dataset are highlighted in italics. In the Table: l-1 phr.is an average number of single-word phrases extracted per text by the algorithm; all phr. is evaluation score when there were no restrictions on keyphrase extraction; multi - evaluation score when algorithms were restricted from extracting single-word keyphrases

The results indicate the following. We observe **two groups of algorithms**. The first group of algorithms tends to minimize the number of extracted single-word keyphrases (TextRank, SingleRank, PositionRank, Topical PageRank). The second group comprises algorithms that extract the highest number of single-word keyphrases (FirstPhrase, TopicRank, MultipartiteRank, KP-miner). Additionally, we highlight two intermediate algorithms: the first, YAKE, is closer and more likely to belong to the first group, while the second, Tfldf, is closer and more likely to belong to the second group of algorithms.

Algorithms from the second group demonstrate the best results compared to other algorithms on datasets with author-assigned keyphrases. However, these algorithms perform worse on datasets with reader-assigned keyphrases. If algorithms from the second group are restricted from extracting single-word phrases, their performance quality improves on datasets with reader-assigned keyphrases. This quality improvement is more noticeable the more pronounced the single-word phrases problem is in the dataset.

For datasets where keyphrases were assigned by readers, algorithms from the first group demonstrated higher performance compared to algorithms in the second group. However, this difference begins to diminish once we prohibit

**Table 3.** Algorithms performance

| DataSet | INSPEC (reader) | | | SemEval2010 (reader) | | | SemEval2017 (reader) | | |
|---|---|---|---|---|---|---|---|---|---|
| Type | l-1 phr. | all phr. | multi | l-1 phr. | all phr. | multi | l-1 phr. | all phr. | multi |
| F@5 | | | | | | | | | |
| FirstPhrase | **2.20** | **25.38** | ***34.57*** | **2.07** | **12.35** | ***14.48*** | **2.33** | **13.71** | **17.38** |
| TextRank | 0.12 | 30.84 | 31.06 | 0.04 | 11.02 | 11.02 | 0.01 | 16.50 | 16.53 |
| SingleRank | 0.33 | 30.72 | 31.83 | 0.30 | 13.69 | *13.73* | 0.17 | 18.78 | 18.46 |
| TopicRank | **2.3** | **25.73** | ***34.17*** | **2.29** | **11.38** | **12.83** | **2.12** | **16.32** | **17.08** |
| MultipartiteRank | **2.22** | **25.92** | ***34.68*** | **2.30** | **12.10** | **13.20** | **2.12** | **16.94** | **17.73** |
| PositionRank | 0.65 | 30.20 | 32.19 | 0.48 | 14.12 | *14.10* | 0.41 | *18.91* | 18.92 |
| TopicalPageRank | 0.42 | 30.48 | 32.00 | 0.33 | 13.60 | 13.67 | 0.17 | *18.80* | 18.56 |
| Tf-Idf | **1.63** | **29.73** | ***34.69*** | **2.15** | **12.32** | ***13.75*** | **1.95** | *18.39* | 17.90 |
| KP-miner | **1.99** | **28.74** | ***34.39*** | **2.36** | **12.12** | **15.04** | **2.61** | **18.34** | ***18.44*** |
| YAKE | 0.82 | 31.96 | 34.04 | 0.95 | *15.12* | *14.65* | 0.55 | *18.36* | 18.18 |
| F@10 | | | | | | | | | |
| FirstPhrase | **4.8** | **30.76** | ***41.92*** | **4.9** | **12.35** | **16.54** | **4.71** | **21.01** | **25.02** |
| TextRank | 1.34 | *37.78* | 40.27 | 0.39 | 15.70 | 15.52 | 0.42 | 24.27 | 24.25 |
| SingleRank | 1.88 | 37.19 | 40.81 | 1.44 | 17.04 | 16.78 | 1.02 | *26.51* | 26.07 |
| TopicRank | **4.97** | **30.20** | **40.29** | **5.18** | **13.51** | **14.57** | **4.85** | **21.83** | **22.70** |
| MultipartiteRank | **4.90** | **31.05** | **41.58** | **5.25** | **14.12** | **15.91** | **4.72** | **22.29** | **24.58** |
| PositionRank | 2.52 | 35.50 | 40.81 | 2.12 | *17.78* | *17.58* | 1.71 | *26.52* | 25.93 |
| TopicalPageRank | 2.03 | 36.98 | 40.80 | 1.55 | 17.14 | 16.96 | 1.14 | *26.66* | *26.26* |
| Tf-Idf | **3.03** | **36.99** | **41.76** | **3.30** | **15.47** | **16.20** | **3.14** | *25.72* | 25.40 |
| KP-miner | **3.44** | **35.78** | **41.73** | **3.66** | **16.41** | **16.90** | **4.33** | **24.96** | **25.36** |
| YAKE | 2.84 | 36.56 | 41.20 | 2.97 | 16.98 | 17.21 | 2.20 | 25.42 | *25.78* |
| Dataset | kp20k (author) | | | PubMed (author) | | | KPBiomed (author) | | |
| F@3 | | | | | | | | | |
| FirstPhrase | **1.05** | *13.91* | 13.64 | **1.29** | *13.84* | 11.20 | **1.20** | *14.10* | 12.66 |
| TextRank | 0.00 | 5.40 | 5.40 | 0.00 | 2.93 | 2.93 | 0.00 | 3.20 | 3.20 |
| SingleRank | 0.04 | 8.69 | 8.59 | 0.04 | 6.54 | 6.21 | 0.04 | 6.11 | 5.99 |
| TopicRank | **1.15** | *12.23* | 11.56 | **1.37** | *12.65* | 9.82 | **1.54** | *12.67* | 10.10 |
| MultipartiteRank | **1.06** | *13.65* | 12.56 | **1.29** | *13.97* | 11.00 | **1.36** | *13.93* | 10.98 |
| PositionRank | 0.09 | 10.86 | 10.61 | 0.11 | 9.96 | 8.87 | 0.07 | 9.34 | 8.80 |
| TopicalPageRank | 0.05 | 9.35 | 9.24 | 0.04 | 7.38 | 6.94 | 0.04 | 6.74 | 6.60 |
| Tf-Idf | **1.03** | *14.17* | 13.03 | **1.47** | *17.33* | *12.20* | **1.60** | *15.13* | 11.90 |
| KP-miner | **1.30** | *14.45* | 14.19 | **1.58** | *17.46* | *13.20* | **1.67** | *15.26* | 13.13 |
| YAKE | 0.24 | 12.65 | 12.09 | 0.31 | 11.83 | 10.36 | 0.35 | 12.04 | 10.46 |
| F@5 | | | | | | | | | |
| FirstPhrase | **1.96** | *15.32* | 13.79 | **2.28** | *16.11* | 12.04 | **2.16** | *17.09* | 13.47 |
| TextRank | 0.04 | 7.93 | 7.91 | 0.00 | 5.19 | 5.19 | 0.01 | 5.01 | 4.97 |
| SingleRank | 0.20 | 11.60 | 11.15 | 0.17 | 9.41 | 8.53 | 0.16 | 8.68 | 8.07 |
| TopicRank | **2.12** | *13.07* | 11.64 | **2.36** | *14.31* | 9.85 | **2.58** | *14.08* | 10.29 |
| MultipartiteRank | **2.04** | *14.31* | 13.11 | **2.28** | *15.72* | 10.96 | **2.41** | *15.75* | 11.83 |
| PositionRank | 0.36 | 13.51 | 12.88 | 0.39 | 13.02 | 10.95 | 0.26 | 12.02 | 10.72 |
| TopicalPageRank | 0.25 | 11.92 | 11.57 | 0.20 | 10.18 | 9.23 | 0.18 | 9.14 | 8.45 |
| Tf-Idf | **1.54** | *14.71* | 12.88 | **2.31** | *18.45* | 11.41 | **2.43** | *16.70* | 12.34 |
| KP-miner | **2.05** | *15.11* | 14.30 | **2.66** | *18.52* | 12.79 | **2.63** | *16.95* | 13.34 |
| YAKE | 0.70 | 14.87 | 13.73 | 0.76 | *14.60* | 11.90 | 0.85 | 15.21 | 12.37 |

algorithms in the second group from extracting single-word phrases.

For datasets with authors – assigned keyphrases, the single-word phrase problem does not exist. Prohibiting the extraction of single-word keyphrases does not enhance the performance quality of algorithms from either group; on the contrary, it degrades their performance.

## 5 Conclusion

The study includes a series of experiments that overcome several shortcomings in unsupervised KE. We formulated the dataset-dependent single-word phrase problem and demonstrated its cause. We analyze how different KE algorithms rank keyphrase candidates and explore one of the reasons that influence quality evaluation.

Results show that an algorithm can perform better or worse depending on whether it allows fewer or more one-word phrases in the extracted keyphrase set. Removing single-word candidate phrases before the ranking stage can improve performance for datasets with reader-assigned keyphrases. This is critical for algorithms that efficiently rank phrases but allow too many single-word phrases among the extracted keyphrases. We also note that prohibiting the extraction of single-word keyphrases does not apply to datasets with authors-assign keyphrases.

## References

1. **Augenstein, I., Das, M., Riedel, S., Vikraman, L., McCallum, A. (2017).** SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, pp. 546–555. DOI: 10.18653/v1/S17-2091.
2. **Bennani-Smires, K., Musat, C., Hossmann, A., Baeriswyl, M., Jaggi, M. (2018).** Simple unsupervised keyphrase extraction using sentence embeddings. Proceedings of the 22nd Conference on Computational Natural Language Learning, pp. 221–229. DOI: 10.18653/v1/k18-1022.
3. **Bernardini, A., Carpineto, C., D'Amico, M. (2009).** Full-subtopic retrieval with keyphrase-based search results clustering. IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, IEEE, pp. 206–213. DOI: 10.1109/wi-iat.2009.37.
4. **Blei, D. M., Ng, A. Y., Jordan, M. I. (2003).** Latent dirichlet allocation. The Journal of Machine Learning Research, Vol. 3, pp. 993–1022.
5. **Boudin, F. (2016).** pke: an open source python-based keyphrase extraction toolkit. Proceedings of the 26th International Conference on Computational Linguistics: System Demonstrations, pp. 69–73.
6. **Boudin, F. (2018).** Unsupervised keyphrase extraction with multipartite graphs. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Vol. 2, pp. 667–672. DOI: 10.18653/v1/N18-2105.
7. **Bougouin, A., Boudin, F., Daille, B. (2013).** TopicRank: Graph-based topic ranking for keyphrase extraction. Proceedings of the 6th International Joint Conference on Natural Language Processing, Asian Federation of Natural Language Processing, pp. 543–551.
8. **Brin, S., Page, L. (1998).** The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems, Vol. 30, No. 1–7, pp. 107–117. DOI: 10.1016/s0169-7552(98)00110-x.
9. **Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., Jatowt, A. (2020).** YAKE! Keyword extraction from single

documents using multiple local features. Information Sciences, Vol. 509, pp. 257–289. DOI: 10.1016/j.ins.2019.09.013.

10. **Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., Jatowt, A. (2018).** A text feature based automatic keyword extraction method for single documents. Springer International Publishing, pp. 684–691. DOI: 10.1007/978-3-319-76941-7_63.
11. **Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., Jatowt, A. (2018).** YAKE! Collection-Independent Automatic Keyword Extractor. Springer International Publishing, pp. 806–810. DOI: 10.1007/978-3-319-76941-7_80.
12. **Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2019).** BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.
13. **Ding, H., Luo, X. (2021).** AttentionRank: Unsupervised keyphrase extraction using self and cross attentions. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 1919–1928. DOI: 10.18653/v1/2021.emnlp-main.146.
14. **Ding, H., Luo, X. (2022).** AGRank: Augmented graph-based unsupervised keyphrase extraction. Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, Association for Computational Linguistics, Vol. 1, pp. 230–239.
15. **El-Beltagy, S. R., Rafea, A. (2010).** KP-miner: Participation in SemEval-2. Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics, pp. 190–193.
16. **Florescu, C., Caragea, C. (2017).** Positionrank: An unsupervised approach to keyphrase extraction from scholarly documents. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Vol. 1, pp. 1105–1115. DOI: 10.18653/v1/p17-1102.
17. **Gutwin, C., Paynter, G., Witten, I., Nevill-Manning, C., Frank, E. (1999).** Improving browsing in digital libraries with keyphrase indexes. Decision Support Systems, Vol. 27, No. 1–2, pp. 81–104. DOI: 10.1016/s0167-9236(99)00038-x.
18. **Houbre, M., Boudin, F., Daille, B. (2022).** A large-scale dataset for biomedical keyphrase generation. Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI), Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Hybrid), pp. 47–53. DOI: 10.18653/v1/2022.louhi-1.6.
19. **Hulth, A. (2003).** Improved automatic keyword extraction given more linguistic knowledge. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Vol. 10, pp. 216–223. DOI: 10.3115/1119355.1119383.
20. **Kim, S. N., Medelyan, O., Kan, M.-Y., Baldwin, T. (2010).** SemEval-2010 task 5 : Automatic keyphrase extraction from scientific articles. Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics, Uppsala, Sweden, pp. 21–26.
21. **Kim, S. N., Medelyan, O., Kan, M. Y., Baldwin, T. (2010).** SemEval-2010 task 5: Automatic keyphrase extraction from scientific articles. Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics, pp. 21–26.
22. **Le, Q. V., Mikolov, T. (2014).** Distributed representations of sentences and documents.

Proceedings of Machine Learning Research, Vol. 32, No. 2, pp. 1188–1196. DOI: 10.48550/ARXIV.1405.4053.

23. **Li, T., Hu, L., Li, H., Sun, C., Li, S., Chi, L. (2021).** TripleRank: An unsupervised keyphrase extraction algorithm. Knowledge-Based Systems, Vol. 219, pp. 106846. DOI: 10.1016/j.knosys.2021.106846.
24. **Liang, X., Wu, S., Li, M., Li, Z. (2021).** Unsupervised keyphrase extraction by jointly modeling local and global context. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 155–164. DOI: 10.18653/v1/2021.emnlp-main.14.
25. **Liu, Z., Huang, W., Zheng, Y., Sun, M. (2010).** Automatic keyphrase extraction via topic decomposition. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 366–376.
26. **Liu, Z., Li, P., Zheng, Y., Sun, M. (2009).** Clustering to find exemplar terms for keyphrase extraction. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 257–266.
27. **Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., Chi, Y. (2017).** Deep keyphrase generation. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vancouver, Canada, pp. 582–592. DOI: 10.18653/v1/P17-1054.
28. **Mihalcea, R., Tarau, P. (2004).** TextRank: Bringing order into text. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 404–411.
29. **Moghadasi, M. N., Zhuang, Y. (2020).** Sent2Vec: A new sentence embedding representation with sentimental semantic. IEEE International Conference on Big Data, IEEE, pp. 4672–468. DOI: 10.1109/bigdata50022.2020.9378337.
30. **Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L. (2018).** Deep contextualized word representations. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Vol. 1, pp. 2227–2237. DOI: 10.18653/v1/N18-1202.
31. **Popova, S., Kovriguina, L., Mouromtsev, D., Khodyrev, I. (2013).** Stop-words in keyphrase extraction problem. Proceedings of the Conference of Open Innovation Association, FRUCT, pp. 113–121. DOI: 10.1109/FRUCT.2013.6737953.
32. **Rose, S., Engel, D., Cramer, N., Cowley, W. (2010).** Automatic keyword extraction from individual documents. Text Mining, pp. 1–20. DOI: 10.1002/9780470689646.ch1.
33. **Schutz, A. (2008).** Keyphrase extraction from single documents in the open domain exploiting linguistic and statistical methods.
34. **Song, M., Xu, P., Feng, Y., Liu, H., Jing, L. (2023).** Mitigating over-generation for unsupervised keyphrase extraction with heterogeneous centrality detection. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 16349–16359. DOI: 10.18653/v1/2023.emnlp-main.1017.
35. **Sun, Y., Qiu, H., Zheng, Y., Wang, Z., Zhang, C. (2020).** SIFRank: A new baseline for unsupervised keyphrase extraction based on pre-trained language model. IEEE Access, Vol. 8, pp. 10896–10906. DOI: 10.1109/access.2020.2965087.
36. **Turney, P. D. (2000).** Learning algorithms for keyphrase extraction. Information Retrieval, Vol. 2, No. 4, pp. 303–336. DOI: 10.1023/a:1009976227802.

**37. Wan, X., Xiao, J. (2008).** Single document keyphrase extraction using neighborhood knowledge. Proceedings of the 23rd National Conference on Artificial Intelligence, AAAI Press, Vol. 2, pp. 855–860.

**38. You, W., Fontaine, D., Barthès, J. P. (2012).** An automatic keyphrase extraction system for scientific documents. Knowledge and Information Systems, Vol. 34, No. 3, pp. 691–724. DOI: 10.1007/s10115-012-0480-2.

**39. Zeng, H. J., He, Q. C., Chen, Z., Ma, W. Y., Ma, J. (2004).** Learning to cluster web search results. , pp. 210–217DOI: 10.1145/1008992.1009030.

**40. Zhu, M., Ahuja, A., Wei, W., Reddy, C. K. (2019).** A hierarchical attention retrieval model for healthcare question answering. The World Wide Web Conference, Vol. 242, pp. 2472–2482. DOI: 10.1145/3308558.3313699.

*Article received on 16/06/2024; accepted on 28/08/2024.*
**Corresponding author is Svetlana Popova.*

ISSN 2007-9737

# Proposal of a Methodology to Identify Unidentified Decedent through Artificial Intelligence Techniques

Ramon A. Briseño[1], Edgar G. Cossio[2,*], Blanca I. Félix[3], Ana C. Ayala[4], Fausto A. Bernal[5], Benjamín I. Hernandez[6], Briseyda P. Martinez[4]

[1] Universidad de Guadalajara,
Centro Universitario de Ciencias Económico Administrativas,
Doctorado en Tecnologías de Información,
Mexico

[2] Instituto de Información Estadística y Geográfica de Jalisco,
Mexico

[3] Instituto Tecnológico Superior de Cajeme,
Mexico

[4] Instituto Tecnológico de Estudios Superiores de Zamora,
Mexico

[5] Universidad Autónoma de Occidente,
Mexico

[6] Instituto Tecnológico de la Piedad,
Mexico

alejandro.bmartinez@alumnos.udg.mx, edgar.cossio@iieg.gob.mx,
18130134@e-itesca.edu.mx, {aayala067, bmartinez077}@accitesz.com,
{z.fausto.a.b.b, ivan1516hc}@gmail.com

**Abstract.** The use of artificial intelligence techniques for the identification and recognition of people is increasing. This article proposes a novel methodology based on artificial neural networks which is based on the classification of tattoos for their analysis and subsequent identification. The techniques used were carefully validated and tested to seek the highest precision in the result. The dataset used was compiled with sample tattoos for training. The model was loaded into a web model that is currently being tested based on Django and python. A dataset and python scripts were built which allowed its implementation.

**Keywords.** Unknown bodies, artificial intelligence, identification, tattoos, neural network.

## 1 Introduction

The existence of a human rights crisis in Mexico is a reality. Day by day, people in their daily lives are exposed to crimes such: as murder, torture, forced disappearance, forced displacement, and executions [1]. This is generating unidentified missing persons. In the state of Jalisco, in 2019, 2,100 people were reported missing [2]. By October that same year, 4,060 unidentified decedents (UD) were reported too [3].

The figures of UD exceed the capacities of forensic institutes; therefore, developing strategies that optimize the early identification of bodies is required [4]. Many times, the features of the bodies can change over time; for example, decomposing bodies of white people change the color of their skin and seem to match individuals of other races. The difficulties of recognizing them increase as time passes since they enter into different changes or cadaveric processes [5, 6].

Identifying and registering dead bodies is an interdisciplinary and complex activity that must be consistent, simple, efficient, and economical. There is no universal identification method of bodies because, in each case, the form of identification that best suits the situation presented must be chosen [4].

However, one of the primary identification methods is comparing the information from the UD (physical characteristics, clothing, artifacts, tattoos, among others) with the information on missing persons [5].

The art of tattooing as a decoration and expression technique has gained importance among many population segments, regardless of age or social class. Figure 1 shows the percentage of male and female bodies with at least one tattoo by age group.

Of the unidentified bodies in Jalisco, 45% had at least one tattoo [4]. It can be identified in Figure 1 that tattoos have a growing trend in the bodies of young UDs. Therefore, creating a tattoo catalog would significantly reduce the time it takes to identify a person.

This work intends to detail a methodology based on artificial vision to compare the information of the missing persons and the UD. The methodology implements a model of convolutional neural networks to recognize and classify tattoos through images and thereby identify missing persons.

This paper is articulated as follows: section two describes the societal problem that represents the identification of unidentified decedents; section three explains the proposed solution to the problem statement using artificial vision technologies; section four explores and discusses the results and, finally, section five concludes and mentions the following steps on this research.

## 2 Problem Statement

Due to the increase in unidentified and unclaimed corpses that have arrived at the Prosecutor's Office facilities since 2006, Forensic Medical Services (SEMEFO as per the acronym in Spanish) presents a saturation of space in its facilities in Jalisco [7].

In forensic identification, there are several methods for the morphological description of corpses, one of which is to identify individual features of each person, such as moles, fractures, and scars, among others. From a forensic point of view, tattoos are interesting because they are "intentional scars" and, at the same time, individualizing features [8].

Forensic identification through tattoos is a method of skin identification currently used by SEMEFO. Said method is effective, efficient, and cheap compared to biological tests [4].

There is a physical catalog, similar to a photo album, where a person searches for a UD among each of the photos until a match is found, the problem is that the search time increases significantly when thousands of bodies have one or more tattoos.

Also, during the identification of a person, the information collected post-mortem and ante-mortem must be as accurate and objective as possible.

Currently, SEMEFO tries to filter the searches by adding descriptions of the tattoos of UD, however, tattoos are art pieces open to personal interpretation, which creates a bias between viewers and the original tattoo [4].

Therefore, it is imperative to optimize the identification process of a UD, which is why this article proposes a methodology with artificial vision

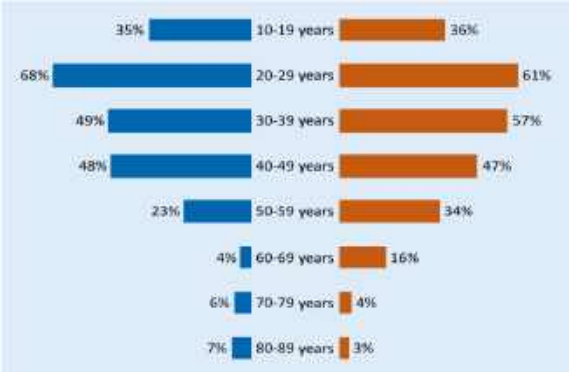



**Fig. 1.** Percentage of bodies with tattoos. The percentage of men (right) and women (left), grouped by age, had at least one tattoo—source: The use of tattoos to identify unknown bodies experiences from Jalisco, Mexico

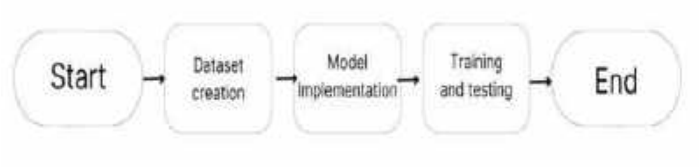



**Fig. 2.** Diagram of the methodology process

**Fig. 3.** Movements to capture a video

techniques to reduce the identification time of a UD.

## 3 Proposal for the Identification of a Body through Tattoos

The identification of people through tattoos can improve with an artificial neural network (ANN). ANNs are computational models based on biological neural networks made up of nodes or neurons, with inputs and outputs simultaneously connected to receiving information [9]. Among them is an image classification field because its main objective is to find behavior patterns from the input data [10].

This paper presents a methodology that implements a convolutional neural network model to identify UD by comparing tattoo images of missing persons with the UD tattoo image database. The methodology consists of three fundamental steps: the elaboration of the dataset (section 3.1), the implementation of the CNN model (section 3.2), and the training and testing of

**Fig. 4.** Tattoos contemplated for this investigation


**Fig 5**. Example of the original image, image without background, and image with the random background

the model (section 3.3). Figure 2 shows the steps of the proposed methodology.

A. Dataset

When the body of a UD arrives at the facilities of a forensic medical services unit, its particular features are identified, including tattoos. In order to use the tattoo classification model through images, the proposal is that a 30-second video is taken of each UD tattoo. Later, with a Python script, decompose the video into each frame. Each frame is saved as a different image of the tattoo.

In addition, for the images generated to be different, the videos are taken by moving the camera at different angles and approaching and retracting it from the tattoo. Figure 3 shows the camera's movements while taking a video of a tattoo.

The dataset is made up of 10 different tattoos voluntarily donated to research. Figure 4 shows the tattoos contemplated in this work.

For this research, a smartphone was used to capture the videos. For each 30-second video, 600 to 900 images were generated. The number of frames per video depends on the camera resolution of the device used to take the video.

In order to increase the number of different images for each tattoo in the data set, two tasks were carried out: 1) the background was removed from each image, leaving a .png image, and 2) a

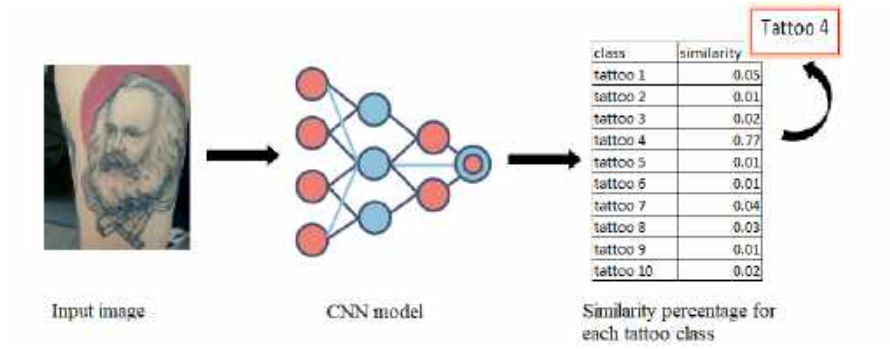



**Fig. 6.** Classification process

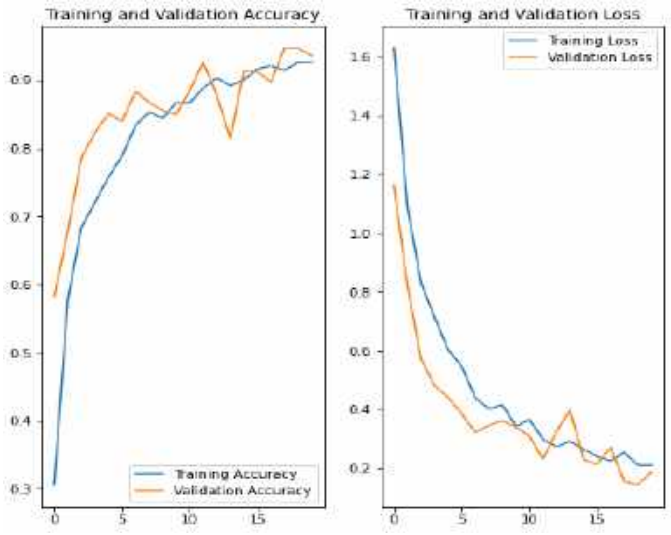



**Fig. 7.** Precision metrics for model training with 20 epochs

random background was added to the images without a background, from a landscape database.

A photo editing software script was used to remove and assign random backgrounds in bulk. Therefore, the data set contains 600 to 900 original images without background and random backgrounds for each of the ten collected tattoos. Figure 5 shows an original tattoo image, an image with no background, and an image with a random background.

B. Convolutional RNA Model

The model uses a Convolutional Neural Network (CNN) to identify UD through tattoo images for image classification. The model used is the MobileNet V2 created by Google ® and previously trained with 1001 classification classes [11].

Due to the use of a pre-trained image classification model, the hidden layers already have preloaded weights in charge of detecting lines, axes, curves, and positions, which improves the classification in the new training of the model (a process called transfer learning).

To use MobileNet V2 in classifying ten tattoos, a Dense layer for ten classes with Softmax activation was added to the model. By freezing the weights of the MobileNet V2 model, said layer allows us to train the model so that it assigns us a percentage of similarity corresponding to each

tattoo in the database. Training that also uses the power of the original Mobile Net V2 model.

To identify a match between a missing person's tattoo and a UD person's tattoo, the model receives an input image and compares it to the classes (UD tattoos) that trained the model. Finally, the model returns an array with the similarity weights for each class. The most prominent similarity percentage represents the tattoo class with the most characteristics in common with the input image. Figure 6, shows the classification process.

The model inputs for training and classification require images with RGB values and a size of 224 x 224 pixels. Since this image size is uncommon in the UD and missing person databases, the images are resized at runtime.

Simultaneously, in each training, the model executes a data augmentation; for each image in the dataset, it creates variations of the same with transformations such as rotation, zoom, translation, and resizing in height and width.

The model executes python scripts to train the model and classify an image using the Keras and Tensorflow libraries.

C. Training and Testing

In training, the data set consisted of two parts: training and validation. The training had 80% of the data set, while the validation had 20%. After multiple trials and errors to find the correct number of training times, the result is that from 20 epochs, the model no longer improves its accuracy.

Therefore, with a training of 20 epochs, the model obtains a 96% accuracy. Figure 7 shows the accuracy of the algorithm over the training periods.

When performing the generalization tests, it was possible to observe that the algorithm classifies well where the conditions of the images were very similar to the training conditions of the algorithm (background colors, contrast, lighting, brightness, zoom, and resolution). Finally, in the cases where the classification failed, the percentage of similarity was found within the five highest.

## 4 Results and Discussion

The model correctly classifies tattoo images in similar conditions to those used in the training process. On the other hand, due to the limitations in the correct classification of images when the input differs significantly from the images used in the training process, its usefulness is reconsidered since it is unlikely that the input images to classify are similar to the images captured from a UD.

However, the model assigns each input image a percentage of similarity for each class in the data set, with which we could generate a top of classes with the highest match.

Although generating a top of matches does not identify in the first instance who the UD is, this could help reduce the search space and shorten the identification times. In this scenario, the most relevant would not be a correct classification by the model but a correct approximation to the classes with more similarity to the input image.

The limitations presented when classifying images very different from those of the data set show that the model continues to present a certain measure of overfitting.

## 5 Limitations

The main limitation is the number of images obtained for each tattoo. Typically, a classification system contains around 1000 different images per class. In the case of UD tattoos, it is impossible to get that many different images.

Even though video decomposition into images generates between 600 to 1000 images per tattoo, these are very similar. To eliminate classification problems such as overfitting caused by the small amount of information, an increase in data was generated in creating the dataset and in each runtime training.

Despite the data augmentation techniques, the model showed limitations when classifying tattoos, where the input image is very different in terms of background colors, contrast, lighting, brightness, and zoom.

## 6 Conclusion and Future Work

The artificial neural network was successfully created and implemented and allowed the identification of people through tattoos. A dataset

was built through which the neural network was trained and tested, resulting in an accuracy of 96%.

In future work, the goal is to expand the tattoo database by taking actual data of UD from forensic institutes as a source. This is in order to evaluate the performance of the model by increasing the number of tattoos in a production scenario.

In addition, it is intended to implement a semantic segmentation technique before classification. This is with the aim of reducing the overfitting caused by the little information collected from each UD tattoo.

The possibility of extending the recognition of people through biometric data and dental records is also being sought.

## References

1. **Franco-Migues, D. (2019).** Tecnologías de esperanza: Apropiaciones tecnopolíticas para la búsqueda de personas desaparecidas en México: El caso de las rastreadoras del fuerte. Comunicación y Sociedad, Vol. 16. DOI:10.32 870/cys.v2019i0.7280.
2. **Martinez, G. (2020).** Jalisco en primer lugar de desapariciones. Tráfico ZMG.
3. **Franco-Migues, D. (2019).** Gobierno de Jalisco lanza sitio de personas fallecidas no identificadas con sólo una quinta parte de los cuerpos que resguardan. ZonaDocs.
4. **Birngruber, C. G., Martinez-Peña, E. G., Corrales-Blanco, L., Holz, F. (2020).** The use of tattoos to identify unknown bodies. Rechtsmedizin, Vol. 30, No. 4, pp. 219–224. DOI: 10.1007/s00194-020-00396-y.
5. **Kumar, A., Harish, D., Singh, A., Kumar, G. A. (2014).** Unknown dead bodies: Problems and solutions. Journal of Indian Academy of Forensic Medicine, Vol. 36, No. 1, pp. 76–80.
6. **Cárdenas-Velarde, J. M., López-Sullaez, L., C. (2019).** Caracterización de cadáveres NN, que fueron ingresados a la morgue judicial, durante la gestión 2015: Protocolo de Identificación de Cadáveres NN. Repositorio Institucional Universidad Mayor de San Andrés, Tesis de Maestría.
7. **Fortuna, M., Corrales, L., Robinson, A., Farias, R. E., Marquez-Grant, N. (2022).** Cuerpos no identificados en el contexto mexicano. Forensic Anthropology, Vol. 5, No. 3. DOI:10.5744/fa.2022.4004b.
8. **Zimmermann, R. (2022).** Quemaduras químicas y carbonización: efectos sobre tatuajes y su incidencia en la identificación de cadáveres. Repositorio Institucional, Universidad Fasta.
9. **Montaño-Moreno, J. J. (2002).** Redes neuronales artificiales aplicadas al análisis de datos. Doctorado Psicología, Facultad de Psicología, Tesis doctoral de la Facultad de Psicología, Universitat de Les Illes Balears, Palma de Mallorca, España.
10. **Domínguez-Gómez, C. (2018).** FacesDetector: Aplicación práctica de machine learning sobre imágenes para un contexto de seguridad (Bachelor's thesis).

*Article received on 01/11/2023; accepted on 17/03/2024.*
**Corresponding author is Edgar G. Cossio.*

ISSN 2007-9737

# Towards a Proto Artificial General Intelligence: The Role of Large Language Model Ontologies in its Development

Christian E. Maldonado-Sifuentes[1], Mariano Vargas-Santiago[2], Samuel Solis-Gamboa[2], Grigori Sidorov[3,*], Luis Lechuga-Gutierrez[2], Francisco González-Andrade[2], María del Carmen Heras-Sánchez[4]

[1] Transdisciplinary Research for Augmented Innovation Laboratory, Mexico

[2] Consejo Nacional de Humanidades Ciencias y Tecnologías, Mexico

[3] Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico

[4] Universidad de Sonora, Mexico

christian.maldonado@trai-l.com, {mariano.vargas, samuel.solis, luis.lechuga, francisco.gonzalez}@conahcyt.mx, sidorov@cic.ipn.mx, carmen@investigacion.uson.mx

**Abstract.** Proto Artificial General Intelligence (ProtoAGI) aims to create a versatile artificial intelligence system capable of autonomously performing diverse tasks. A foundational element of ProtoAGI is the Large Language Model (LLM) ontology, which plays a crucial role in organizing and retrieving information about different LLMs, enabling the selection of the most appropriate model for specific tasks. This ontology, the first of several designed to support ProtoAGI, addresses key challenges in managing and accessing information regarding LLM capabilities, performance, and task suitability. We present the methodology for constructing this ontology, covering data extraction, enrichment, and model recommendation using a generalized LLM API. The initial version of this ontology involved processing over a million tokens, underscoring the system's complexity and the scale of information integrated. This ontology is designed for continuous updates, ensuring that ProtoAGI remains current with the latest advancements in LLMs. The ongoing development of this ontology marks a significant step in ProtoAGI's evolution, following an initial proof-of-concept demonstrated during the 2024 eclipse, where the feasibility of integrating such a comprehensive LLM ontology into a general-purpose AI system was shown. By making this ontology accessible to the broader AI community, we aim to accelerate further advancements in AGI research and applications.

**Keywords.** Artificial general intelligence, large language models, ontology, hybrid intelligent systems, multi-agent systems.

## 1 Introduction

The aspiration to develop Artificial General Intelligence (AGI) systems with human-like cognitive capabilities across a diverse array of tasks stands at the forefront of artificial intelligence research. AGI seeks to transcend the specialized functionalities of current AI technologies, aiming instead for a breadth and adaptability of intelligence that mirrors human cognition. From the perspective of research the development of AGI also represents a pivotal goal, aimed at creating systems that emulate the

comprehensive cognitive capabilities of humans across varied tasks.

AGI aspires to surpass the specialized functions typical of contemporary AI technologies, striving for a level of versatility and adaptability akin to human intellect. This ambition extends beyond the replication of human-like problem-solving and learning capacities; it also seeks to incorporate creativity, emotional intelligence, and an understanding of complex contexts and nuances. In this manuscript, we present and advancement over our original proof-of-concept:

Proto Artificial General Intelligence (ProtoAGI), a cutting-edge AGI simulation system marked by its dynamic learning capabilities and a modular, scalable architecture. ProtoAGI signifies a departure from conventional AI systems, embodying a framework designed for continual evolution and adaptation.

Central to this system is a self-enhancing knowledge base that facilitates iterative learning and reflection, thereby redefining the standards of machine cognition's flexibility and depth. We introduce a general ontology within ProtoAGI, enabling the system to effectively address and resolve diverse problems.

This ontology functions as a sophisticated mechanism that discerns and categorizes inputs, ensuring the delivery of precise and contextually appropriate outcomes. The incorporation of this ontology not only enriches the system's operational intelligence but also enhances its capability to perform across both general and specialized tasks with heightened accuracy. Our approach underscores the necessity of developing AGI systems that are not only technically proficient but also ethically informed and capable of seamless integration within human-centric environments.

The architecture of ProtoAGI is strategically crafted to propel forward the domain of AGI by tackling critical challenges such as learning efficiency, adaptability, and comprehension of human emotions and ethical considerations. Recent advancements in AGI research underscore its growing relevance and potential applications across various sectors. Studies by Naudé and Dimitri in 2020 highlight the implications of AGI for public policy, discussing the control and political challenges arising from its development [5].

Fei et al. (2022) explore a multimodal foundation model for AGI, trained on extensive data to handle various cognitive tasks, marking a significant step toward realizing AGI [3]. Yamakawa (2021) adopts a whole brain architecture approach, aiming to accelerate AGI development by mimicking brain structures and functions [12].

Additionally, Shevlin et al. (2019) discuss the inherent limitations and challenges in achieving AGI despite advancements in machine intelligence [8]. These contributions not only position ProtoAGI within the contemporary research landscape but also illuminate the system's potential to advance AGI technologies that are adaptable, ethically responsible, and capable of complex, human-like reasoning and learning.

In a previous manuscript, we introduced ProtoAGI, a novel AGI simulation system characterized by its dynamic learning capabilities and a modular, scalable architecture. The Large Language Model (LLM) ontology is a critical component of ProtoAGI, organizing and retrieving relevant information about various LLMs, enabling the selection of the most suitable model for specific tasks. This article focuses on the importance of the LLM ontology, the methodology used to develop it, and the iterative nature of its creation, marking significant progress towards completing ProtoAGI's first prototype.

## 2 Related Works

Research in Artificial General Intelligence (AGI) aims to develop systems with cognitive capabilities that mirror human intelligence, transcending the specialized functionalities of current AI technologies. Significant contributions to this field include the development of computational cognitive architectures and the integration of symbolic and sub-symbolic computations. The ACT-R computational cognitive architecture is a notable example, offering a framework for modeling human-like cognitive abilities.

This architecture has been validated against extensive experimental data, showcasing its potential in creating systems with general

intelligence capabilities akin to those of humans [4]. Furthermore, the integration of symbolic and sub-symbolic computation is pivotal for advancing human-level artificial intelligence. This approach addresses the complex challenge of developing systems that can operate across varied tasks with adaptability and depth akin to human cognition [2].

These studies underline the importance of a holistic approach in AGI research, combining technical proficiency with sophisticated cognitive models to foster systems capable of complex reasoning and learning.

Research on Artificial General Intelligence (AGI) with human-like cognitive capabilities has been progressing with significant contributions from various scholars. Rayhan et al. [7] provide a roadmap to achieving human-level capabilities in AGI, highlighting the developmental strides necessary for such advancements. This work lays a foundational perspective on the scope and ambition of AGI research.

Tong et al. [10] explore the incorporation of affective dimensions into AGI through neural-symbolic computing. Their work aims to endow artificial systems with a more nuanced understanding of human emotions, which is crucial for real-life interactions and cognitive processes.

Jovanović [11] discusses the current limitations and barriers that prevent the realization of AGI. His analysis identifies key areas where further research and technological development are needed, providing a critical overview of the field's challenges. Sonko et al. [9] offer a critical review of the ethical considerations and challenges associated with the development of AGI. Their insights are vital for guiding future research towards responsible and ethically informed technological advancements.

Azam et al. [1] introduce a novel model of narrative memory for conscious agents, aiming to enhance the memory capabilities of AGI systems to better simulate human-like cognitive functions. This work contributes to the broader goal of creating more sophisticated and versatile AGI systems. Together, these studies underscore the diverse approaches and ongoing research efforts aimed at achieving AGI with human-like cognitive abilities, reflecting both the potential and the hurdles in the path towards such advanced artificial intelligence systems.

Despite these advancements, a significant gap in current AGI research is the absence of a comprehensive, general ontology that could unify various approaches and facilitate the development of truly generalizable AGI systems.

Existing works often focus on specific aspects or functionalities, lacking an overarching framework that integrates different cognitive models and computational architectures. Such a general ontology would be instrumental in standardizing methodologies, promoting interoperability between systems, and accelerating progress towards achieving genuine AGI. Without it, efforts remain fragmented, and the vision of fully realizing AGI's potential remains elusive.

## 3 Importance of LLM Ontology

The LLM ontology is essential for several reasons:

- **Organization of Information**: It systematically organizes information about various LLMs, including their capabilities, specializations, and computational requirements.
- **Facilitating Model Selection**: It aids in selecting the most suitable model for a specific task by providing detailed information and recommendations based on the model's characteristics.
- **Enhancing Efficiency**: By automating the model selection process, it reduces the time and effort required to identify the optimal LLM, thereby improving overall efficiency.
- **Continuous Improvement**: The ontology can be continuously fetched and updated, ensuring that the latest models and their attributes are always available for selection.

### 3.1 Optimizing ProtoAGI with LLM Ontologies

ProtoAGI is designed to be a versatile AI system capable of autonomously performing a wide range of tasks. A crucial component of this system is the Large Language Model (LLM) ontology, which plays a pivotal role in organizing and retrieving information about various LLMs.

This allows the system to select the most suitable model for a given task. The LLM ontology serves as the first of several planned ontologies that will systematically provide accessible knowledge to ProtoAGI.

The LLM ontology addresses a core challenge: Managing and accessing detailed information about different models, their capabilities, and their performance across diverse contexts. This section outlines the methodology used to create the ontology, which includes data extraction, enrichment, and model recommendations using a generalized LLM API.

#### 3.1.1 Problem Representation

- **Ontologies**:
  - $\mathcal{L}$: Set of LLMs, $\mathcal{L} = \{L_1,\ L_2,\ \ldots,\ L_m\}$.
  - $\mathcal{A}$: Set of algorithms, $\mathcal{A} = \{A_1,\ A_2,\ \ldots,\ A_n\}$.
  - $\mathcal{D}$: Set of datasets, $\mathcal{D} = \{D_1,\ D_2,\ \ldots,\ D_p\}$.
- **Selection Variables**:
  - $x_{ijk} \in \{0, 1\}$: Binary variable indicating whether LLM $L_i$, algorithm $A_j$, and dataset $D_k$ are selected together.
- **Objective Function**:
  - Maximize $f(x_{ijk})$.
  - Here, $f(x_{ijk})$ is a function that evaluates the performance of the combination of $L_i$, $A_j$, and $D_k$ for a specific task.
- **Constraints**:
  - Each task requires exactly one combination of $L_i$, $A_j$, and $D_k$:

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{p} x_{ijk} = 1. \tag{1}$$

  - Additional constraints based on resource limits, compatibility, etc:

$$g(x_{ijk}) \leq R, \tag{2}$$

    where $g(x_{ijk})$ represents resource usage (e.g., computation, memory), and $R$ is the resource limit.

#### 3.1.2 Combinatorial Space

The combinatorial space involves all possible combinations of LLMs, algorithms, and datasets. The total number of combinations is $m \times n \times p$. A brute force approach would involve evaluating every combination:

$$\text{Total combinations} = m \cdot n \cdot p. \tag{3}$$

While this guarantees finding the global optimum, the computational cost becomes prohibitive as $m$, $n$, and $p$ grow large, making it impractical for large-scale systems like ProtoAGI.

### 3.2 Advantages of LLM-guided Selection over Brute Force Search

Instead of a brute force search, ProtoAGI leverages multiple specialized LLMs for their reasoning capabilities, enabling them to intelligently guide the search for optimal combinations within the ontology.

This method offers several advantages over brute force search by balancing the trade-off between precision and computational efficiency, effectively replacing human-in-the-loop processes with AI-driven decision-making.

#### 3.2.1 Brute Force Search

- **Total combinations**: $m \times n \times p$
- **Objective function evaluation**:

$$f(L_i, A_j, D_k) \quad \forall \quad \begin{matrix} i \in \{1, \ldots, m\}, \\ j \in \{1, \ldots, n\}, \\ k \in \{1, \ldots, p\}. \end{matrix} \tag{4}$$

- **Deterministic precision**: Evaluates every possible combination, guaranteeing the optimal solution.
- **Computational cost**: Extremely high for large $m$, $n$, and $p$, making this approach inefficient and infeasible for large-scale tasks.

#### 3.2.2 LLM-guided Selection

- **Total combinations**: Reduced to a manageable subset $\mathcal{S}$.
- **Stochastic approximation**:

$$\tilde{f}(L_i,\ A_j,\ D_k) \text{ for selected } (i,\ j,\ k) \in \mathcal{S}, \tag{5}$$

where $\mathcal{S} \subseteq \{1, \ldots, m\} \times \{1, \ldots, n\} \times \{1, \ldots, p\}$.

- **Simulated reasoning capabilities**: LLMs use their knowledge and contextual understanding to predict the most promising combinations.
- **Human-in-the-loop replacement**: LLMs streamline the decision-making process by emulating expert reasoning.

#### 3.2.3 Mathematical Representation

- **Brute Force Search**:

  Optimal combination: $= \max_{i,j,k} f(L_i,\ A_j,\ D_k)$ with:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} 1 = m \cdot n \cdot p. \tag{6}$$

- **LLM-guided Selection**:
  - **Subset selection**: LLMs reduce the search space to a subset $\mathcal{S}$.
  - **Stochastic approximation**:

    Approximate optimal combination $= \max_{(i,j,k) \in \mathcal{S}} \tilde{f}(L_i, A_j, D_k)$ .
  - **Reduced search space**:

$$|\mathcal{S}| \ll m \cdot n \cdot p. \tag{7}$$

#### 3.2.4 Advantages of LLM-guided Selection

- **Efficiency**:
  - LLMs drastically reduce the number of combinations to evaluate, improving computational efficiency.
  - The search space is reduced from $O(m \cdot n \cdot p)$ to $O(|\mathcal{S}|)$, where $|\mathcal{S}|$ is much smaller.
- **Simulated Reasoning**:
  - LLMs utilize prior knowledge and contextual understanding to predict promising combinations, thus avoiding unnecessary evaluations.
  - Provides a probabilistic assessment of potential solutions, leading to quicker convergence to high-quality solutions.
- **Stochastic Approximation**:
  - Balances precision with computational efficiency by approximating the optimal solution.
  - Enables adaptive and dynamic decision-making, similar to expert human evaluation and refinement.
- **Human-in-the-loop Replacement**:
  - LLMs reduce reliance on manual decision-making processes, enhancing scalability and reducing human error.
  - Improves repeatability in selecting optimal combinations, especially for complex or large-scale tasks.

ISSN 2007-9737



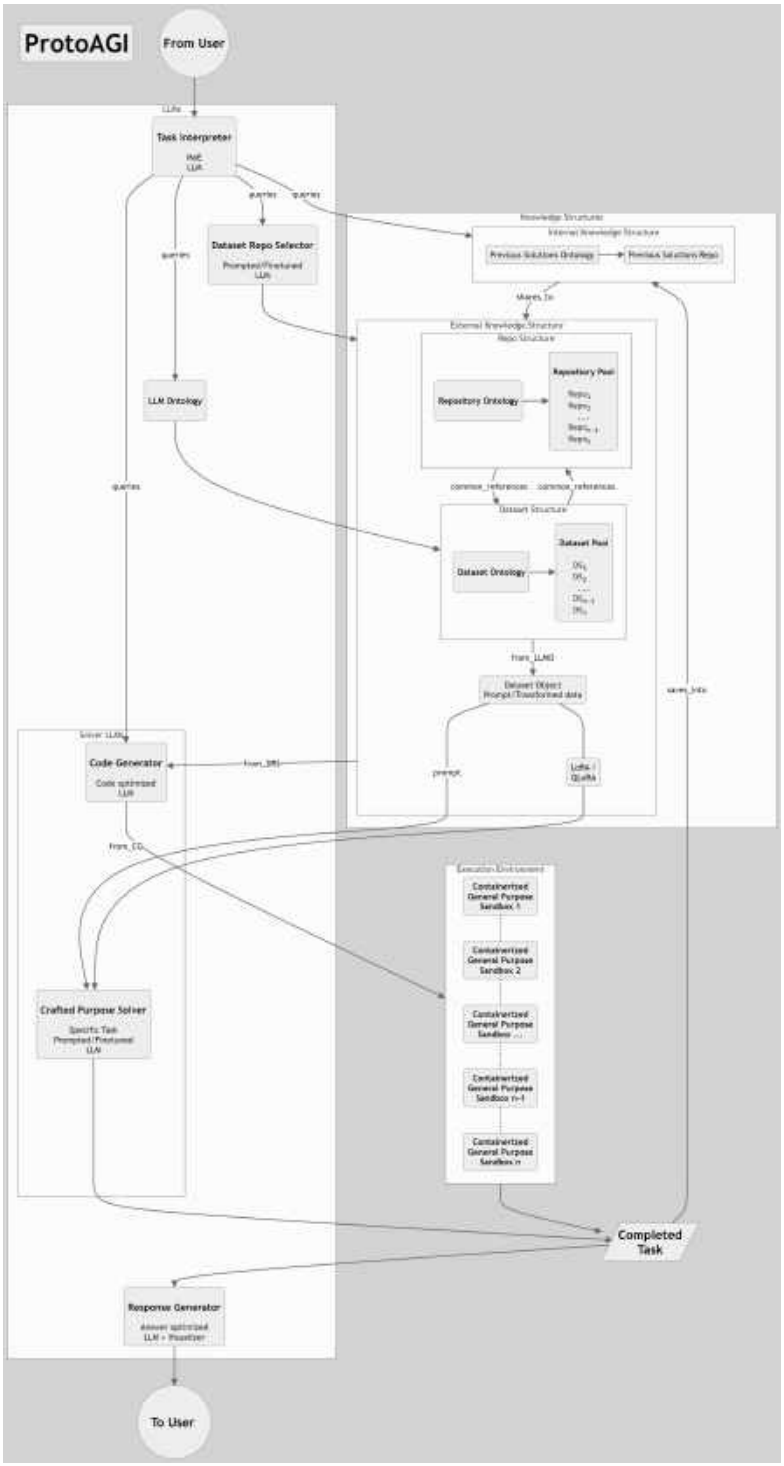



**Fig. 1.** ProtoAGI system overview

#### 3.2.5 Stochastic Approximation Model

Given the LLM's ability to simulate reasoning, its selection process can be modeled as follows:

– **Initial Predictions**:

$$P(x_{ijk} = 1) = \text{LLM}_{\text{score}}(L_i, A_j, D_k). \quad (8)$$

– **Thresholding**:

$$\mathcal{S} = \{(i, j, k) \mid \text{LLM}_{\text{score}}(L_i, A_j, D_k) \geq \tau\}, \quad (9)$$

where $\tau$ is a threshold score to filter top candidates.

– **Evaluation**:

$$\tilde{f}(L_i, A_j, D_k) \quad \forall (i, j, k) \in \mathcal{S}. \quad (10)$$

– **Selection**:

$$\text{Optimal combination} \approx \max_{(i,j,k)\in\mathcal{S}} \tilde{f}(L_i, A_j, D_k). \quad (11)$$

## 4 Our Method

ProtoAGI is a versatile AI system designed to autonomously perform a wide range of tasks by leveraging multiple specialized Large Language Models (LLMs). A critical component of ProtoAGI is the LLM ontology, which organizes and retrieves relevant information about various LLMs to facilitate the selection of the most suitable model for specific tasks. This section details the methodology behind the creation of the LLM ontology, highlighting its integration within the ProtoAGI system.

### 4.1 ProtoAGI System Overview

#### 4.1.1 Dynamic Learning Capabilities

ProtoAGI signifies a departure from conventional AI systems, embodying a framework designed for continual evolution and adaptation. Central to this system is a self-enhancing knowledge base that facilitates iterative learning and reflection, thereby redefining the standards of machine cognition's flexibility and depth.

#### 4.1.2 Modular and Scalable Architecture

ProtoAGI is characterized by its modular, scalable architecture that allows for seamless integration of various AI components. This design ensures that the system can adapt to new tasks and incorporate advancements in AI technologies.

#### 4.1.3 General Ontology Integration

A general ontology within ProtoAGI enables the system to effectively address and resolve diverse problems. This ontology functions as a sophisticated mechanism that discerns and categorizes inputs, ensuring the delivery of precise and contextually appropriate outcomes.

#### 4.1.4 Ethical and Human-centric Design

Our approach underscores the necessity of developing AGI systems that are not only technically proficient but also ethically informed and capable of seamless integration within human-centric environments. ProtoAGI is strategically crafted to tackle critical challenges such as learning efficiency, adaptability, and comprehension of human emotions and ethical considerations.

### 4.2 Ontology Creation Method

The process of creating the LLM ontology involves several steps, including data extraction, ontology enrichment, and model recommendation. This first iteration involved processing over a million tokens to achieve the final version.

#### 4.2.1 Data Extraction and Parsing

The first step involves extracting relevant information about different LLMs from a specified HTML source, namely from the Ollama repository [6]. Ollama serves not only as a comprehensive database but also as a framework for running LLMs in a containerized environment similar to Docker. This ensures consistent performance and ease of deployment across different systems.

The extraction process is achieved using a PHP script that parses the HTML and retrieves key details such as the model's name, URL, description, and additional metadata like the number of pulls, tags, and the last update time. This data is then encoded into a JSON structure for further processing.

#### 4.2.2 Ollama Repository

The Ollama repository is a comprehensive database of large language models, providing detailed information about each model's capabilities, usage statistics, and other relevant metadata. The repository serves as a central hub for researchers and developers to access a wide variety of LLMs. Additionally, Ollama functions as a framework to run these models in isolated, containerized environments, similar to Docker, which facilitates easy deployment and scalability.

By utilizing the Ollama repository, we ensure that our ontology is built upon a robust and diverse dataset, encompassing the latest advancements in LLM technology. This rich source of data allows us to create a well-rounded and informative ontology that can support a wide range of applications within the ProtoAGI system. The extraction process from the Ollama repository involves several steps:

1. **HTML Parsing**: Using PHP, we parse the HTML content of the Ollama repository to identify and retrieve relevant sections containing LLM information.
2. **Data Retrieval**: Extract key details such as the model's name, URL, description, number of pulls, tags, and the last update time.
3. **JSON Encoding**: Encode the retrieved data into a structured JSON format for further processing and enrichment.

By leveraging the extensive data available in the Ollama repository, we ensure that our LLM ontology is comprehensive and up-to-date, reflecting the latest trends and capabilities in the field of large language models.

#### 4.2.3 Tag Extraction and Ontology Enrichment

After extracting the basic information, we enrich our ontology by retrieving additional tags associated with each model from their respective URLs. This process involves fetching the HTML content of the tag pages and extracting the relevant tags using DOM and XPath queries. The enriched data is then encoded into a JSON structure. This step ensures that the ontology captures a comprehensive set of attributes for each model.

#### 4.2.4 LLM Integration for Model Recommendations

To provide detailed descriptions and implications of each model and its associated tags, we integrate an LLM API. This API is used to generate detailed explanations and recommendations for each model based on its description and tags.

The API queries are designed to slow down request handling to avoid overloading the system. This process involved processing over a million tokens to ensure the accuracy and comprehensiveness of the final recommendations. The overall implementation involves three main steps:

1. **HTML Parsing and Data Extraction**: Using PHP, we parse the HTML content to extract models' information, including their names, descriptions, URLs, and metadata.
2. **Tag Retrieval and Ontology Enrichment**: Additional tags are retrieved from specific URLs, and the ontology is enriched with this information.

3. **Model Recommendations Using LLM API**: Detailed descriptions and recommendations are generated for each model and its tags using the LLM API.

The combination of these steps allows our system to effectively select the most suitable LLM for a given task, considering the model's specialization, computational requirements, and other relevant factors. The continuous fetching and updating mechanism ensures that the ontology remains current and relevant.

### 4.3 Leveraging Previous Solutions Ontology

As ProtoAGI evolves, we can leverage a previous solutions ontology to enhance efficiency. This ontology stores past successful configurations of LLMs, algorithms, and datasets, allowing the system to refer to these solutions and reduce computational requirements.

#### 4.3.1 Improved Performance Over Time

The performance of ProtoAGI improves over time as more solutions are added to the previous solutions ontology. Let $t$ represent time and $C(t)$ represent the cumulative computational cost up to time $t$. The gradient of improved performance, $G(t)$, can be modeled as:

$$G(t) = \frac{\partial P(t)}{\partial t}, \tag{12}$$

where $P(t)$ is the performance function, which improves as more solutions are added. The computational cost $C(t)$ for each new solution is: $C(t) = O(1)$ for retrieval $+O$ (evaluation cost) for adding a new solution.

#### 4.3.2 Diminishing Computational Power Requirements

The presence of a previous solutions ontology reduces the need for extensive computation. The system can quickly refer to past successful configurations, minimizing the search space and computational cost. The cost of using the previous solutions ontology is:

$$\text{Previous solutions ontology cost: } O(1). \tag{13}$$

Combining this with our LLM-guided selection approach, the overall computational cost becomes:

$$\text{Optimized cost: } O(|\mathcal{S}|) + \sum_{i=1}^{N} O(\text{evaluation cost}_i), \tag{14}$$

where $|\mathcal{S}| \ll m \cdot n \cdot p$ and $N$ is the number of solutions added to the ontology. As $N$ increases over time, the retrieval cost remains $O(1)$, but the evaluation cost for adding new solutions gradually decreases due to the growing efficiency of the ontology.

In this way, ProtoAGI benefits from both efficient LLM-guided selection and the reduced computational requirements provided by the previous solutions ontology, with performance improving as more solutions are added.

### 4.4 Continuous Fetching and Updating

The LLM ontology is designed to be continuously fetched and updated, reflecting the dynamic nature of the AI landscape. As new models are developed and existing models are updated, the ontology can incorporate these changes, ensuring that ProtoAGI always has access to the most up-to-date information. This capability is crucial for maintaining the system's relevance and effectiveness in selecting the best LLM for a given task.

#### 4.4.1 First Iteration and Token Utilization

The first iteration of the LLM ontology involved an extensive process of data processing and enrichment, requiring the use of over a million tokens. This significant utilization of tokens was essential for capturing the comprehensive details and nuances of each model, ensuring both accuracy and depth in the ontology.

#### 4.4.2 Data Processing

To begin with, the initial data extraction phase involved parsing large amounts of HTML content from the Ollama repository. This repository contains a vast array of information on various LLMs, including their descriptions, usage statistics, and metadata such as tags and last update times. The PHP script used in this phase had to efficiently handle and process this extensive dataset to extract relevant information accurately.

#### 4.4.3 Token Utilization for Data Enrichment

After the initial extraction, the next phase focused on data enrichment. This step was crucial for adding more context and detailed descriptions to each LLM entry. By utilizing a significant number of tokens, we were able to:

- Generate comprehensive descriptions that cover various aspects of each LLM, including their capabilities, specializations, and ideal use cases.
- Extract and include additional metadata such as tags, which help in categorizing and understanding the specific features and functionalities of each model.
- Ensure that the data is structured in a way that facilitates easy retrieval and application within the ProtoAGI system.

#### 4.4.4 Ensuring Robustness and Scalability

The intensive token usage was not merely for immediate accuracy but also to lay a solid foundation for future iterations of the ontology. By creating a detailed and well-organized initial dataset, we ensured that the ontology is robust and scalable. This means that as new LLMs are developed and added to the Ollama repository, the ontology can be updated and expanded without compromising its integrity or performance.

#### 4.4.5 Continuous Improvement and Updates

Given the rapidly evolving nature of AI and LLM technologies, it is imperative that the ontology remains current. The foundational work done in the first iteration supports continuous updates and improvements. As new models are introduced and existing models are updated, the ontology can seamlessly incorporate these changes, maintaining its relevance and utility for the ProtoAGI system.

## 5 Results

The resulting Large Language Model (LLM) ontology is structured as a comprehensive JSON file, consisting of a total of **20,283 elements**. This file provides a well-organized, machine-readable repository of information regarding various LLMs, their capabilities, specializations, and metadata, designed to facilitate the selection of the most suitable models for specific tasks. The full ontology can be accessed at the following address[1].

### 5.1 Structure of the JSON File

The JSON file encodes detailed information about each LLM, which is organized in a modular structure for efficient access and retrieval. The key fields included in the ontology are as follows:

- **Model Name**: A string representing the unique name of the LLM.
- **Description**: A comprehensive description of the LLM, highlighting its key capabilities, intended use cases, and performance characteristics.
- **URL**: A link to additional information about the LLM, including documentation and source repositories.
- **Metadata**:
  - **Pulls**: An integer value representing the number of times the model has been pulled or used from its repository, providing an indication of its popularity and utilization.

[1] trai-l.com/AIG/parser/JSON/processed_llm_ontology.json

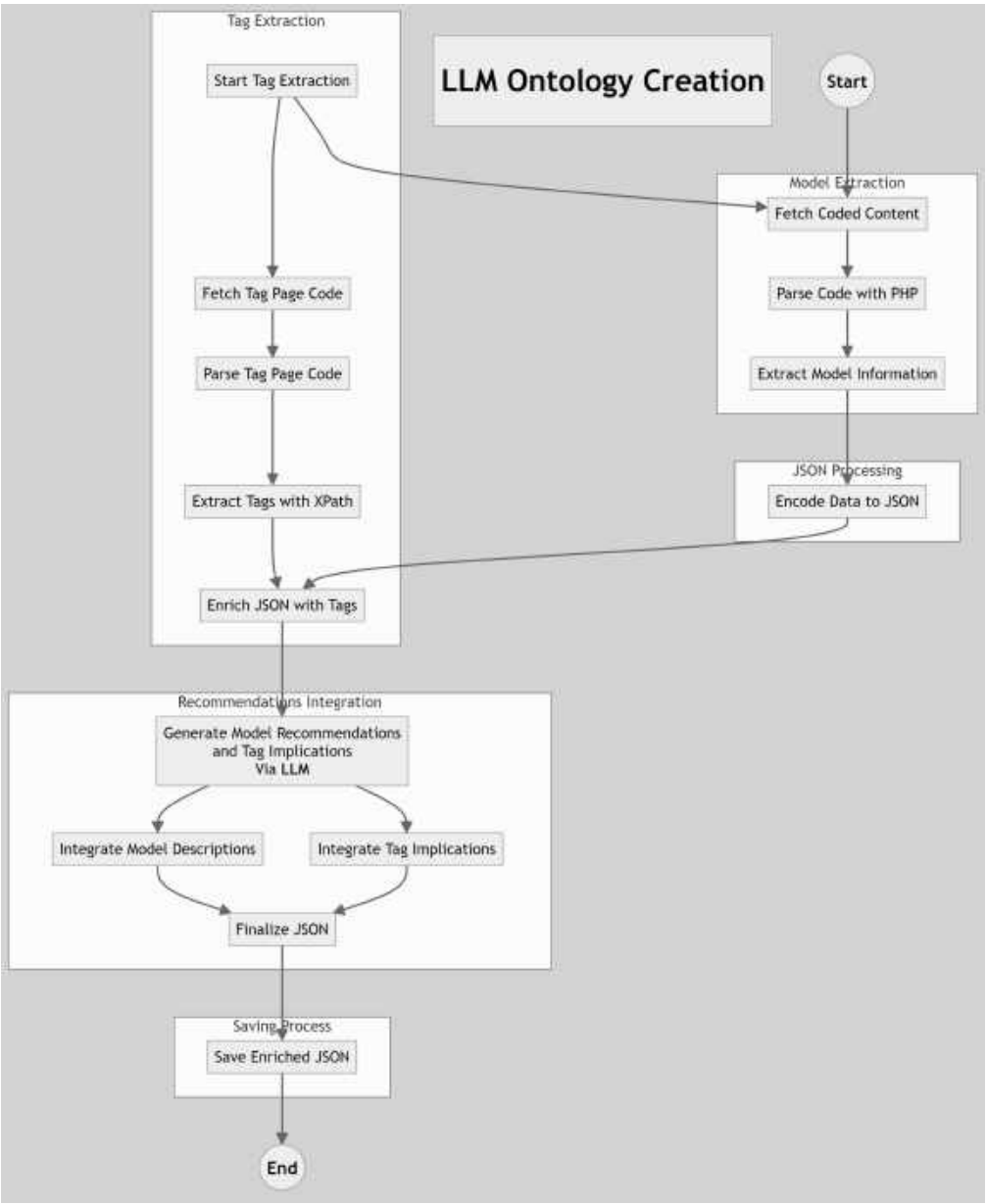



**Fig. 2.** Detailed flowchart for LLM ontology creation

– **Tags**: A list of categorical tags associated with the model, each providing insights into its functionalities, such as “NLP”, “Text Generation”, “Multimodal”, etc. This allows for efficient filtering and categorization.

– **Last Update**: A timestamp that specifies the last update time of the model, ensuring users are informed of the model’s most recent state.

– **Tag Descriptions**: A set of detailed explanations for each tag associated with the model, outlining the implications and relevance of each tag to the model’s performance or

intended applications.

### 5.2 Processing and Token Utilization

The first iteration of this ontology involved extensive data processing, using over **1 million tokens** to capture the detailed information and metadata for each LLM. The process included:

– **Data extraction** from multiple sources, including model repositories and documentation.

– **Ontology enrichment** through the incorporation of additional metadata, such as tags and usage statistics.

– **Model recommendation generation** using an LLM API, which provided descriptions and tag implications based on extracted data.

This large-scale token processing ensured comprehensive coverage of each LLM's attributes, enabling the ontology to serve as a robust foundation for automating LLM selection within the ProtoAGI system.

### 5.3 Technical Details of the Ontology

With 20,283 total elements, the ontology covers a broad range of LLMs across various categories, providing detailed descriptions and metadata for each. The large number of elements allows for fine-grained model selection and efficient retrieval of information tailored to diverse AI tasks. The following summarizes the content:

– **LLMs covered**: The ontology includes models for natural language processing (NLP), text generation, multimodal systems, and other AI applications.

– **Metadata depth**: Each model entry contains multiple layers of metadata, providing insights into model performance, popularity, and applicability in different contexts.

– **Extensibility**: The ontology is designed for continuous updates, allowing new models and tags to be added with minimal changes to the underlying structure.

This extensive effort ensures that the ontology provides a scalable, dynamic, and comprehensive repository, laying the groundwork for efficient model selection and recommendation within the ProtoAGI system.

## 6 Discussion

The creation of an efficient LLM ontology introduces a transformative approach to addressing some of the most critical challenges in Artificial General Intelligence (AGI) research, especially concerning the exponential growth of computational requirements. As AI models continue to increase in complexity, the need for scalable and efficient resource management becomes paramount.

The LLM ontology developed for ProtoAGI significantly contributes to this goal by optimizing the model selection process, thereby reducing the computational overhead traditionally associated with brute force search methods. One of the primary contributions of the LLM ontology is its ability to intelligently manage computational resources. By curating a structured knowledge base of LLMs, including metadata such as model performance, capabilities, and resource consumption, the ontology enables ProtoAGI to make informed decisions when selecting models for specific tasks. This ensures that only the most appropriate models are chosen, avoiding redundant or resource-intensive computations.

This model selection optimization drastically reduces the computational load, making it a critical tool for scaling AGI systems. In the context of supercomputing, the LLM ontology becomes even more important. As we move toward the next generation of AI systems, the role of supercomputers in training and deploying massive AI models will become more prevalent.

Supercomputers equipped with hundreds of thousands of cores and GPUs are uniquely positioned to handle the computational demands of AGI systems. However, even with access to such immense computational power, resource efficiency remains vital.

The ontology enables ProtoAGI to take full advantage of supercomputing infrastructure by

selecting models that best balance performance and resource use, ensuring that computational power is not wasted on suboptimal models. This efficiency allows AGI systems to scale effectively, leveraging supercomputers to perform vast numbers of parallel computations while maintaining high performance.

Additionally, the previous solutions ontology further amplifies the system's efficiency by allowing ProtoAGI to learn from its prior configurations. This repository of successful model-algorithm-dataset combinations allows the system to bypass unnecessary evaluations of previously tested solutions, significantly reducing both the search space and computational cost.

In a supercomputing environment, where every computation may consume significant time and energy, such optimizations are crucial for maintaining system performance and throughput. The LLM ontology's efficiency improvements are not just a technical advancement—they carry broader implications for the scalability and deployment of AGI systems in real-world applications.

As AGI moves closer to widespread deployment across sectors such as healthcare, finance, autonomous systems, and scientific research, it is essential that these systems are capable of operating efficiently in constrained environments, where computational resources are either limited or highly expensive. The ability to dynamically adjust model selection based on the available resources ensures that AGI systems remain adaptable, making them more practical for both research and industry.

Supercomputing also plays a critical role in accelerating AGI development. Training large-scale LLMs, simulating neural networks with human-like cognitive abilities, and processing immense datasets all require computational resources that far exceed the capacity of traditional computing environments.

The LLM ontology provides a pathway for maximizing the utility of supercomputing infrastructure by reducing the resource requirements for both training and inference tasks. As AI models and AGI systems grow larger, more intricate, and more reliant on massive datasets, the role of efficient resource allocation, powered by supercomputers, will become increasingly central to the success of AGI.

Moreover, the continuous updating capability of the LLM ontology ensures that ProtoAGI stays current with the latest advancements in LLMs, datasets, and algorithms. As new models are developed and existing models are improved, the ontology can be updated dynamically, ensuring that ProtoAGI remains relevant and at the cutting edge of AGI research. This continuous integration of new data not only improves ProtoAGI's model selection but also helps the system evolve as the field of AI progresses.

From a broader perspective, the LLM ontology has profound implications for the general AI and machine learning communities. By providing a structured, continuously updated repository of LLM capabilities, it enables researchers and developers to leverage this resource for a wide range of applications. The ontology can support everything from educational tools and research projects to industrial-scale AI deployments.

Its ability to offer insights into the strengths and weaknesses of existing models can foster innovation and collaboration across various AI domains, leading to the development of more specialized and efficient AI models. Finally, as AI research moves forward, the integration of supercomputing with intelligent, resource-efficient systems like ProtoAGI will be critical.

The LLM ontology provides a template for how supercomputing resources can be managed more effectively, ensuring that as AGI systems scale, they can maintain efficiency, reduce energy consumption, and maximize performance. This is essential for enabling AGI to fulfill its potential across a wide array of sectors, while ensuring that the ever-increasing computational demands do not become a bottleneck for future advancements.

## 7 Conclusion

In conclusion, the development of the LLM ontology marks a significant advancement in the ProtoAGI system. By systematically organizing and enriching information about various LLMs, this ontology facilitates the efficient selection of the

most suitable models for specific tasks. This process not only improves the overall efficiency of the ProtoAGI system but also ensures that it remains adaptable and up-to-date with the latest advancements in AI models.

The LLM ontology also serves as a valuable resource for the broader AI community. Its structured and detailed repository of LLM information can be utilized in various contexts, including educational purposes, research, and practical applications in industry. By providing a reliable and continuously updated source of model capabilities, the ontology supports the development and deployment of advanced AI systems across multiple domains.

Overall, the development of the LLM ontology represents a significant step towards realizing the vision of AGI. By addressing the critical challenges of model selection and resource management, this ontology provides a robust framework for advancing AGI technologies that are both efficient and adaptable, ultimately contributing to the broader goal of achieving human-like cognitive capabilities in artificial systems. The benefits of this ontology extend beyond ProtoAGI, offering valuable insights and resources that can be leveraged by the entire AI community to foster further advancements in the field.

## References

1. **Azam, M., Rafiq, T., Gul-Naz, F., Ghafoor, M., Un-Nisa, M., Malik, H. (2024).** A novel model of narrative memory for conscious agents. International Journal of Information Systems and Computer Technologies, Vol. 3, No. 1, pp. 12–22. DOI: 10.58325/ijisct.003.01.0080.
2. **Besold, T. R., Lamb, L., Icard, T. F., Miikkulainen, R. (2015).** Proceedings of the 10th international workshop on neural-symbolic learning and reasoning.
3. **Fei, N., Lu, Z., Gao, Y., Yang, G., Huo, Y., Wen, J., Lu, H., Song, R., Gao, X., Xiang, T., Sun, H., Wen, J. R. (2022).** Towards artificial general intelligence via a multimodal foundation model. Nature Communications, Vol. 13, No. 1. DOI: 10.1038/s41467-022-30761-2.
4. **Lebiere, C. (2011).** Profile: The FMS cognitive modeling group at carnegie mellon university. The IEEE Intelligent Informatics Bulletin, Vol. 12, No. 1, pp. 1–3.
5. **Naudé, W., Dimitri, N. (2019).** The race for an artificial general intelligence: implications for public policy. AI and Society, Vol. 35, No. 2, pp. 367–379. DOI: 10.1007/s00146-019-00887-x.
6. **Ollama (2024).** Large language models repository. https://ollama.com/library.
7. **Rayhan, A., Rayhan, R., Rayhan, S. (2023).** Artificial general intelligence: Roadmap to achieving human-level capabilities. DOI: 10.13140/RG.2.2.33680.79361/1.
8. **Shevlin, H. (2019).** The limits of AGI: Challenges and perspectives.
9. **Sonko, S., Adewusi, A. O., Obi, O. C., Onwusinkwue, S., Atadoga, A. (2024).** A critical review towards artificial general intelligence: Challenges, ethical considerations, and the path forward. World Journal of Advanced Research and Reviews, Vol. 21, No. 3, pp. 1262–1268. DOI: 10.30574/wjarr.2024.21.3.0817.
10. **Tong, J. C. H., Hsu, Y. F., Liau, C. J. (2024).** An exploring study on building affective artificial intelligence by neural-symbolic computing. Proceedings of the AAAI Symposium Series, Vol. 3, No. 1, pp. 592–593. DOI: 10.1609/aaaiss.v3i1.31288.
11. **Voss, P., Jovanovic, M. (2023).** Why we don't have AGI yet. DOI: 10.48550/ARXIV.2308.03598.
12. **Yamakawa, H. (2021).** The whole brain architecture approach: Accelerating the development of artificial general intelligence by referring to the brain. Neural Networks, Vol. 144, pp. 478–495. DOI: 10.1016/j.neunet.2021.09.004.

ISSN 2007-9737



*Article received on 11/05/2024; accepted on 09/07/2024.*
**Corresponding author is Grigori Sidorov.*

ISSN 2007-9737

# Multi-document Text Summarization through Features Relevance Calculation

Verónica Neri-Mendoza, Yulia Ledeneva*,
René Arnulfo García-Hernández, Ángel Hernández-Castañeda

Autonomous University of the State of Mexico,
Toluca, State of Mexico,
Mexico

veronica.nerimendoz@gmail.com, yledeneva@yahoo.com,
renearnulfo@hotmail.com, anhernandezc@uaemex.mx

**Abstract.** Multi-document text summarization is obtaining relevant information from a set of documents describing the same topic. However, determining the key sentences in the text to be presented as a summary is difficult. Consequently, it is necessary to use features that help to identify informative sentences from those that are not. However, distinguishing between significant and insignificant features is a challenging task. In this study, we introduced a method to assess the impact of 19 linguistic and statistical features derived from human-written reference summaries. Moreover, we tested them using the DUC01 dataset in two lengths (50 and 100 words). The results demonstrate that the proposed method outperforms state-of-the-art approaches and heuristics based on the ROUGE-1 metric.

**Keywords.** Text features, summarization, multi-document, contribution of features.

## 1 Introduction

Written news is one of the most important forms of expression for citizens to know and understand real-world events. Hundreds of news stories are generated every day, causing information overload. Because of this, it would be easier for readers to read representative fragments of a set of news stories than to read each [29, 20].

Due to various research, it is known that the task of summarization combines important reading and writing skills, as well as the understanding of a large amount of linguistic knowledge [20, 15]. Automatic Text Summarization (ATS) involves extracting the most essential information from a document or a set of documents using advanced methods [2, 24, 23, 7]. There are different classifications of ATS according to how the summary is generated. There are [9, 1, 18]:

**Abstractive:** This approach produces summaries incorporating new content using external resources to interpret the source code.

**Extractive:** In this approach, summaries are produced by weighing sentences to assign a value to each sentence and then selecting the highest values.

**Hybrid:** Summaries are produced by combining the advantages of abstractive and extractive approaches.

Based on the quantity of input documents, summarization techniques can be classified as Single-Document or Multi-Document approaches (ATSMD) [9].

To summarize a text, humans follow the next steps: Read the text, underline the main ideas, and rewrite the main ideas [20, 15]. Commonly, ATS: Commonly, the ATS involves calculating the

relevance of each sentence through text features and selecting the $k$ sentences with the best relevance as a summary [9, 29].

Research in extractive ATS has explored various features to identify text segments that capture the main idea of a document set. These features are categorized into statistical and linguistic types. Statistical features focus on the distribution of words or topics without interpreting the content of the document, while linguistic features involve applying linguistic knowledge to analyze sentence structures [9, 18].

However, the following questions remain open:

1. What is the contribution of text features?
2. Which sentences will be included?

With this in mind, we examined 19 different statistical and linguistic features and computed the relevance coefficient for each one to assess its contribution. Furthermore, to select the most important sentences, we used a genetic algorithm (GA) to maximize the weight of sentences. Moreover, we tested the proposed method at two different levels of compression: 50 and 100 words. The remainder of the paper is structured as follows: Section 2 reviews the related literature, Section 3 outlines the proposed method, Section 4 presents the experimental results, and Section 5 provides the conclusions.

## 2 Related Works

The effectiveness of features relies on their application and combination to assess the importance of each sentence in the source documents. Assessing the contribution of text features aids in creating a more accurate summary. The two questions mentioned in Section 1 have been addressed as exposed in Sections 2.1 and 2.2.

### 2.1 What is the Contribution of Text Features?

State-of-the-art methods take various approaches to determining the contribution of text features. Some methods evaluate the sentences within input documents and assign a relevance score to each feature.

#### 2.1.1 Scoring from source documents

The importance of features is established from the source documents, with weights assigned to each feature based on the text content. For example, in [14], a straightforward yet competent method for generating summaries through term frequency was introduced. Term frequency generally serves as a criterion for identifying more relevant sentences. The sentences are subsequently ranked based on their scores.

While in [4], the position of sentences and word frequency were initially considered for summarization. Later, in [8], additional text features like key terms and similarity to the title were introduced.

#### 2.1.2 Scoring through coefficient optimization

In [27], sentence extraction was achieved by generating combinations of relevance coefficients. Initially, these coefficients were assigned randomly within the range of 0 to 1, and were subsequently refined using the GA.

In [10], a GA was employed to determine the optimal set of relevance coefficients for ten features, including sentence position, similarity to the title, presence of named entities, and sentence length. The impact of each feature was initially studied to facilitate summary generation. Subsequently, the features were used to train a GA and a mathematical regression algorithm to determine the optimal set of text features and relevance coefficient values.

In the studies analyzed in this section, sentences were evaluated using relevance coefficients, which are integrated into the sentence score by applying the fitness function outlined in Equation 1:

$$F(s) = \sum_{j=1}^{C} f_j(s), \tag{1}$$

where, $f_j$ is the relevance coefficient for each $jth$ feature, $s$ is the sentence score, and $C$ is the total number of features.

#### 2.1.3 Scoring from manual coefficients

Similar to how relevance is determined based on coefficients computed through optimization, text feature coefficients have also been calculated manually.

In [19], a method was introduced that combines semantic and statistical features, such as key sentences, sentence length, presence of proper nouns, sentence position, similarity to the title, sentence centrality, and inclusion of numbers. During the sentence selection stage, sentences were evaluated using a linear sum, with the coefficients manually determined.

In [16], To evaluate the quality of a summary, an ensemble of features that are both domain—and language-independent was used. These features included similarity to the title, sentence position, sentence length, cohesion between sentences, and coverage. The features were optimized using a memetic algorithm.

In [25], a GA was proposed for generating summaries by selecting sentences using four features: coverage, sentence position, sentence length, and similarity to the title. The results showed enhancements in sentence selection. Nevertheless, the coefficients were manually determined based on the assumption that these values would improve sentence selection. Consequently, these approaches depend on subjective criteria for setting the coefficients.

### 2.2 Which Sentences will be Included?

Generating a summary is a crucial step. The chosen features and their relevance coefficients determine which sentences most effectively describe the document. Various techniques have been applied to this stage in the literature, including decision trees, lexical chains, clustering, latent semantic analysis, neural networks, and optimization methods.

Each of these techniques has its limitations. Clustering is straightforward and intuitive but limits elements to being assigned to one group [5, 9].

Graph-based methods offer understandable models for representing documents but involve complex construction and storage, and they may not accurately capture the definition of words or sentences [9]. On the other hand, deep learning methods, while effective, need extensive training data [9, 28, 27]. Latent semantic analysis-based methods depend heavily on the grade of the semantic representation of the source documents [5].

Decision trees can only detect sentence associations based on shared phrases [9, 5]. Therefore, it is crucial to determine the contribution of features by deriving weighting coefficients through methods that balance the quality of the summary with the cost of its generation. In ATS research, several datasets, including human-written reference summaries, have been developed to assess the performance of proposed methods. The aim is for the software-produced summaries to be similar to those created by humans.

Despite research into relevance coefficients through optimization or manual assignment, there has been a lack of investigation into using human-written reference summaries as an objective standard for calculating these coefficients in the current state of the art.

## 3 Proposed Method

Given the uncertainty about the usefulness of calculating the contribution of statistical and linguistic text features based on human-written reference summaries, we propose a methodology comprising the following steps: Calculating Text Features, Calculating Relevance Coefficients, concatenating and pre-processing source documents, then performing feature extraction and sentence selection.

### 3.1 Calculating Text Features

The input for this process consisted of human-written reference summaries. These documents were preprocessed through normalization, text derivation, and removal of stopwords. The source documents were tagged with Parts-of-Speech (POS) and Named Entity Recognition (NER) tags. Besides, the content was vectorized using the Word2vec word embedding model to capture word meanings and enhance the linguistic concepts of

the sentences [11]. After this, we consider the following text features:

### 3.1.1 Inclusion of Thematic Words (TW)

TW pertains to topic particular words that frequently appear in the content. In the proposed method, we evaluated rate values ranging from 5% to 15%. Empirically, we observed that using 7% of the most common words. Those that would give a general overview of the documents could be extracted. This feature was calculated using the equation 2:

$$TW(s) = \frac{Number\ of\ TW \in s}{Total\ Number\ of\ TW}, \quad (2)$$

where the weight of thematic words $(TW)$ of sentence s is equal to the number of thematic words in the sentence $s$ divided by the total number of thematic words in the document

### 3.1.2 Inclusion of Positive Keywords (PW)

Given that words are the essential components of a sentence, a sentence with more content keywords is considered more important. Therefore, we established positive keywords as the top 7% of the most recurrent words in the documents, as this percentage effectively identifies thematic words. The weight of this feature was calculated using Equation 3:

$$PW(s) = \frac{1}{length(s)} \sum_{i=1}^{n} tf_i \cdot P(s \in D|PW), \quad (3)$$

where $PW(s)$ is the sum of the frequencies of each term considered as a keyword multiplied by its respective probability value in the sentence $s$, which belongs to the document $D$.

### 3.1.3 Inclusion of Title Words (ITW)

Sentences that contain words from the title may be indicative of the topic of the document and are more likely to be included in the abstract. For this reason, the sentence obtains a high score if it includes words that show in the title of the document. This feature was calculated using equation 4:

$$ITW(s) = \frac{Words\ \in s \cap Words\ \in\ T}{Length(T)}, \quad (4)$$

where $ITW(s)$ = Is obtained by the intersection of the $words$ that belong to the sentence $s$ and the $words$ that belong to the title $(T)$, between the $length$ of the $T$

### 3.1.4 Inclusion of POS and NER Tagging

The presence of POS or NER tags can indicate the importance of words in a sentence. While it is possible to capture the frequency of all available POS or NER tags (54 in total), we focused on the most common ones (14), which are listed in the following table:

**Table 1.** Description of POS and NER tags

| Tag | Description | Tag | Description |
|---|---|---|---|
| CC | Conjunction | DT | Determiner |
| CD | Cardinal number | JJ | Adjective |
| VB | Verb base form | IN | Preposition |
| NN | Singular noun | NNS | Plural noun |
| VBD | Verb in the past | PER | Personal |
| VBN | Past participle | DATE | Periods |
| GPE | Cities and states | ORG | Organizations |

The contribution of this feature was calculated using the term frequency of tagged words (see equation 5):

$$TF(t_i) = \frac{t_i}{N}, t_i \in \{POS, NER\}. \quad (5)$$

### 3.1.5 TF-IDF

Term Frequency (TF) estimates how often a word is included in a source document, while Inverse Document Frequency (IDF) considers the number of sentences in which the word appears. A higher TF-IDF value indicates that the word is more frequent in the sentence but less common across the document (see Equation 6):

$$TF - IDF(s) = TF \times (w) log(\frac{N}{L}), \quad (6)$$

where $TF$ = number of times word $w$ occurs in the document. $N$ = Total number of documents in the corpus and, $L$ = Total number of documents in which the word $w$ occurs.

TF-IDF has the following properties. It assigns a weight to the word $w$ in document. It is the highest when $w$ occurs many times within a document and does not occur in the rest of the documents in the corpus. It is lower when $w$ occurs fewer times in a document or occurs in many documents. It is the lowest when w occurs virtually in all the documents.

#### 3.1.6 Main sentence similarity (SMS)

This feature evaluates the similarity between a sentence $s$ and the main sentence of the document ($MS$). The use of centrality grows diversity. For this reason, we proposed that the sentence with the highest score in the $TF - IDF$ calculation be the main sentence, $MS$. We computed its similarity to other sentences using cosine similarity and Word2vec vectorization (see equation 7):

$$SMS(s, MS) = \frac{\sum_{i=1}^{n} s_i SMS}{\sqrt{\sum_{i=1}^{n} s_i}\sqrt{\sum_{i=1}^{n} MS_i}}. \quad (7)$$

### 3.2 Calculating Relevance Coefficients

This step aims to identify the relevance of each feature by computing coefficients based on human-written reference summaries.

Starting from the calculation of the features described above, the following steps were carried out:

1. A feature matrix was created for each human-written reference document. In this matrix, the columns represent the calculated values for each feature $(C_i)$, while the rows correlate with the sentences from the source document.
2. The scores obtained by each feature in the document were summed $\sum_{i=1}^{n} C_i$.
3. The average of each feature from human-written reference summaries was computed ($\frac{\sum(C_i)}{d}$), where $d$ is the number of human-written reference summaries.
4. The relevance coefficients for each feature were calculated from the earlier averages using Bayesian probability. This probability is favorable as it allows the designation of probabilities to individual events and allows the calculation of an event probability based on known probabilities of related events. Equation 8 describes how the relevance coefficients ($w_i$) are computed:

$$relevance\ (w_i) = \frac{\frac{\sum_{i=1}^{n} C_i}{d}}{\sum_{j=1}^{m} \frac{\sum_{i=1}^{n} C_i}{d}}, \quad (8)$$

where $j$ represent the $j^{th}$ text feature and $m$ is the number of human reference summaries. As the output of this process, we obtained a vector of relevance coefficients with values ranging from 0 to 1.

### 3.3 Multi-document Summarization Process

This process is initiated with a collection of news documents that need to be summarized (also called source documents). For each collection of source documents, the following processes are applied:

1. **Pre-processing:** The order of the source documents must represent the chronological sequence of events. Therefore, the news in the collection was combined hierarchically to create a meta-document organized from the oldest to the newest news.

   Afterward, the text was normalized through lemmatization, and stopwords were filtered. Moreover, we applied POS and NER tagging. Lastly, the sentences were vectorized using Word2vec to capture word meanings and construct linguistic concepts.

2. **Sentence selection:** The GA was employed to assess and enhance the selection of sentences that will form the summary, like a combinatorial optimization problem.

   This process emulates evolution, gradually and repeatedly refining the given target objective. The ”strongest” solutions persist, while the ”weakest” ones are eliminated [11]:

- **Encoding:** Binary, a gene means a sentence, and the individuals represent candidate summaries.
- **Initial population:** Randomly.
- **Operators:** Selection, crossover, and mutation operators were applied to obtain new solutions.
- **Text Features:** They were computed according to the equations shown in section 3.1. Then, a feature matrix was created. In this matrix, the columns depict to the score values $S_i$ of features, while the rows correlate with the sentences in the candidate summaries. Since the $S_i$ scores, a vector of scores was generated employing the equation: $(\sum S_i)$.
- **Fitness Function:** The candidate summaries were assessed by equation 9:

$$FA = Max \sum (\sum s_i \times w_i). \quad (9)$$

  The fitness of the candidate document $D_i$ is the maximization of the linear sum of the scores acquired $\sum s_i$ multiplied by their relevance coefficients $W = w_i$ (vector obtained in 3.2 section).
- **Stop criteria:** Number of generations.

## 4 Experimental Results

To know the performance of the proposed method. The tests were implemented under the DUC01 dataset. This dataset serves as a point of reference for estimating the quality of summaries and comprises 309 documents ordered in 30 collections. It focuses on English-language news articles. The dataset includes two human-written reference summaries for assessment [1].

Two summaries were created for each collection with compression rates of 50 and 100 words. Moreover, we analyzed the contribution of a group of 19 linguistic and statistical features calculated from the human-written reference summaries.

The contribution percentages of these features in the generated summaries are shown in Fig. 1. It is observed that the inclusion of thematic words (TW) in the summaries makes a more significant contribution (35%) because when they appear frequently in the documents, they are related to the topic addressed. Consequently, they make the sentence informative.

In contrast, the singular (NN) and plural noun (NNS), personal names (PERSON), organizations (ORG), as well as names of countries, cities, and places (GPE) were features that contributed to the generation of the summaries with, a sum of 21% between them. These features are important because they capture information about who or what about is in the sentence.

One of the important parts of sentences are the determiners (DT), which, together with the verb, personal pronouns or the subject, help give meaning and context to the nouns since they add information about quantity and possession, which is why they represent 4% of relevance in the generation of the summaries.

Adjectives (JJ) that express characteristics or properties attributed to a noun contributed 5%. At the same time, the actions performed by the nouns were captured by the verbs in base form (VB), the verb in the past (VBD), and the past participle (VBN) with a sum of 5% between them.

Regarding the structure of the summaries, the grammatical categories coordinating conjunction had a contribution of 2%, while the prepositions and conjunctions (IN) contributed 5% of relevance. Due to is necessary because they allow the creation of relationships between words and sentences. As for dates (DATE), 2% of relevance was included, while for the cardinal numbers (CD) feature, the contribution was 3% since it can reflect transactions and percentages. The idea is if a sentence contains numerical data, it is important and very likely to be included in the summary.

Finally, Fig. 1 also shows that with regard to TF-IDF, the contribution was 8% relevance. This feature was used to identify the most distinctive thematic features of the documents. In addition, Fig. 1 shows the similarity of the sentence with the main sentence (SMS) contributed 9% of relevance. Finally, the inclusion of title words (ITW) and positive keywords (PW) contributed 2% of relevance each.

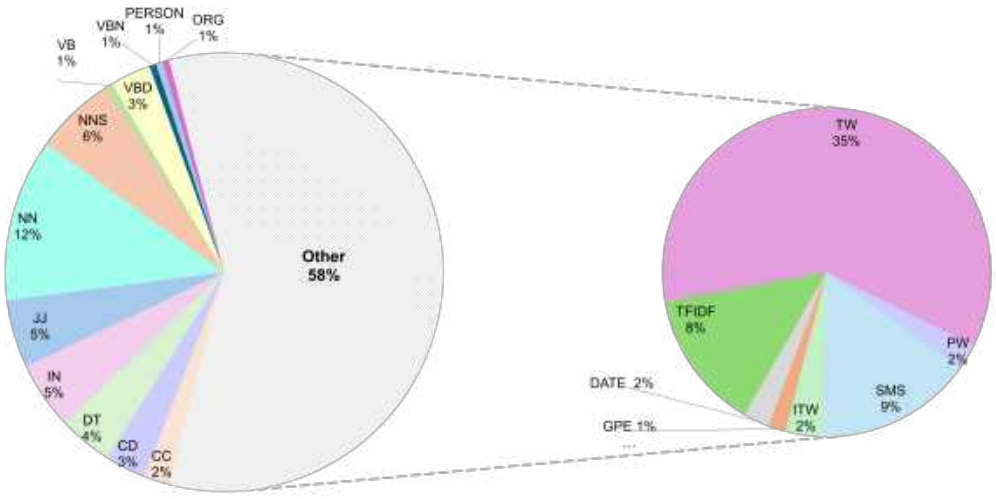



**Fig. 1.** Percentage distribution of contribution of text features

The GA parameters used to select the sentences that formed the final summaries are shown in the next table:

**Table 2.** Parameters of GA

| Feature | 50 words | 100 words |
|---|---|---|
| Generations | 15 | 85 |
| Population size | $2 \times number\ of\ sentences$ | |
| Elitism | 0.03% | |
| Selection operator | Roulette | |
| Inversion mutation | 0.009% | |

As observed in Table 2, the number of generations varied according to the summary length. The longer the length, the more generations were required. Moreover, the best results were obtained with the selection operator roulette operator.

We utilized the ROUGE system to assess the summaries produced by the proposed method. This system measures the quality of the generated summaries by comparing them with human-written reference summaries using n-grams. Specifically, we emphasized using ROUGE-1 and ROUGE-2, which are widely regarded as a dependable metric for this type of evaluation [7].

The heuristics used to contrast the performance of the proposed method are outlined below.

**Topline:** Consists of obtaining the best selection of sentences (via GA) according to their similarity concerning human-written reference (ideal) summaries. Therefore, these summaries are a reference point that any ATSMD method aspires to achieve, even if there is disagreement among ideal summaries [21].

**BF:** The Baseline-first (BF) selects the first sentences from source documents sorted chronologically, generating extractive summaries according to the number of words.

**BR:** The Baseline-random (BR) randomly selects sentences from source documents till the required length is complete to include them as a summary.

**BFD:** Baseline-first-document (BFD) takes out the first sentences from the earliest document until the required summary length is reached.

**LB:** Lead Baseline (LB) incorporates the first 50 and 100 words of the most recent document as a summary. Likewise, the input documents must be chronologically sorted.

**Table 3.** Comparison with heuristics and state-of-the-art methods 50 Words

| Method | ROUGE-1 | Advance (%) | ROUGE-2 | Advance (%) |
|---|---|---|---|---|
| Topline | 40.395 (1) | 100.000 % | 15.648(1) | 100.000 % |
| GA | 28.023 (2) | 39.258% | 6.272 (2) | 31.656 % |
| Proposed | 27.854 (3) | 38.427% | 4.699 (3) | 20.190 % |
| RBM | 27.369 (4) | 36.046% | 4.617 (4) | 19.593% |
| BFD | 25.435 (5) | 26.551% | 4.301 (7) | 17.289% |
| BF | 25.194 (6) | 25.368% | 4.596 (5) | 19.440% |
| Baldwin | 22.906 (7) | 14.134% | 3.054 (8) | 8.200% |
| CBA | 22.679 (8) | 13.020% | 2.859 (10) | 6.778% |
| LB | 22.620 (9) | 12.730% | 4.341 (6) | 17.581% |
| NeATS | 22.594 (10) | 12.603% | 2.963 (9) | 7.536% |
| BR | 20.027 (11) | 0.000% | 1.929 (11) | 0.000% |

**Table 4.** Comparison with heuristics and state-of-the-art methods 100 Words

| Method | ROUGE-1 | Advance (%) | ROUGE-2 | Advance (%) |
|---|---|---|---|---|
| Topline | 47.256 (1) | 100.000 %) | 18.994(1) | 100.000 % |
| Proposed | 34.053 (1) | 34.838 % | 7.632 (2) | 27.708% |
| GA | 33.985 (2) | 34.503 % | 7.617 (3) | 27.613% |
| RBM | 32.923 (3) | 29.261 % | 6.985 (4) | 23.592% |
| BF | 31.716 (4) | 23.304 % | 6.962 (5) | 23.445% |
| BFD | 30.462 (5) | 17.115 % | 5.962 (6) | 17.083% |
| Baldwin | 28.647 (6) | 8.158 % | 4.760 (7) | 9.435% |
| NeATS | 28.195 (7) | 5.927 % | 4.037 (9) | 4.835% |
| LB | 28.195 (8) | 5.927 % | 4.109 (8) | 5.293% |
| BR | 26.994 (9) | 0.000 % | 3.277 (11) | 0.000% |
| CBA | 26.741 (10) | -1.248% | 3.510 (10) | 1.482% |

Subsequently, we present the state-of-the-art techniques used to compare the performance of the proposed method.

**CBA:** The Clustering-Based Approach (CBA) creates summaries using sentences as topics. The topics are then clustered using two types of clustering: hierarchical and partitioning (K-means). Finally, the most relevant topics are selected for the final summary [6].

**NeATS:** NeATS is a method that employs term clustering (also known as the "buddy system") to match sentences to select the most relevant sentences from source documents [12].

**GA:** The authors in [17] proposed a GA to optimize sentence selection using Coverage and Sentence position.

**RBM:** This method proposes using the Restricted Boltzmann Machine (RBM) to identify the relationships among nine text features. These features include TF-IDF, SMS, POS, NER, and Sentence Length [26].

**Baldwin:** This method employs sentence selection using entropy [3]. Therefore, a sentence concerning the collection of documents is relevant if it contains words of low entropy.

Tables 3 and 4 compare the proposed method with state-of-the-art techniques and heuristics.

**Table 5.** Resulting ranking of the methods

| Method | Position | | | | | | | | | | | Result Rank |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | |
| Topline | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4.000 (1) |
| Proposed | 2 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3.636 (2) |
| GA | 2 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3.636 (2) |
| RBM | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2.909 (3) |
| BFD | 0 | 0 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 2.181 (5) |
| BF | 0 | 0 | 0 | 0 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 2.454 (4) |
| Baldwin | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 1 | 0 | 0 | 0 | 1.727 (6) |
| CBA | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 2 | 1 | 0.818 (9) |
| LB | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 1.454 (7) |
| NeATS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | 1 | 0 | 1.090 (8) |
| BR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 0.454 (10) |

Additionally, we calculated the improvement in the summarization task, considering that any method cannot perform worse than randomly selecting sentences (BR), which is set to 0%.

The best possible performance, referred to as the Topline, is set at 100%. By utilizing BR and the Topline, we can recalculate the F-measure results to assess the improvement relative to the worst and best results. This advancement is displayed in the third and fifth columns of the tables. The number in parentheses within each table slot indicates the ranking of each method.

The ROUGE-1 and ROUGE-2 scores demonstrate that the proposed method surpasses all state-of-the-art methods and heuristics for the lengths of summaries of 100 words. Although the proposed method does not outperform the GA method in 50 words summaries, it achieves a comparable value, indicating a promising gap for future research. Additionally, the method enhances sentence selection overall. Furthermore, the proposed method shows close performance to the Topline, highlighting the extent of the achieved improvement.

To consolidate all the results from ROUGE-1 and ROUGE-2 for 50 and 100 words, Table 5 presents them in a unified format, ranking them based on Equation 10, which has been applied in [17]:

$$Rank\ (method) = \sum_{r=1}^{11} \frac{(11 - r + 1)R}{11}, \qquad (10)$$

where $R$ indicates the number of times the method appears in the r-th rank.

Table 5 provides a comprehensive comparison of the summarization methods. From the findings, we can note that the BR demonstrates the lowest performance.

Meanwhile, both the proposed method and the GA significantly enhance results, achieving second place in the ranking.

## 5 Conclusions

In existing literature, human-written reference summaries have typically been used to evaluate the performance of proposed methods, not to determine the score of the features.

Our findings indicate that thematic words are the most influential feature in summary generation, with features related to nouns, verbs, and adjectives also playing significant roles. Additionally, we evaluated the contribution of features related to grammatical categories, such as determiners, conjunctions, and prepositions.

After determining the contribution of each feature, we optimized sentence selection using GA. The results demonstrate an enhancement in sentence selection across various summary lengths, as indicated by the ROUGE-1 and ROUGE-2 measures.

The contribution derived from human-written reference summaries offers a valuable starting point for assigning relevance to features, offering practical insights for future research and development. However, since ROUGE depends on human-written summaries for evaluation, it is crucial to assess the performance of methods using evaluation techniques that do not rely on human references [13, 22].

## References

1. **Akhmetov, I., Nurlybayeva, S., Ualiyeva, I., Pak, A., Gelbukh, A., Akhmetov, I., Nurlybayeva, S., Ualiyeva, I., Pak, A., Gelbukh, A. (2023).** A comprehensive review on automatic text summarization. Computación y Sistemas, Vol. 27, No. 4, pp. 1203–1240.
2. **AL-Khassawneh, Y. A., Hanandeh, E. S. (2023).** Extractive Arabic text summarization-graph-based approach. Electronics 2023, Vol. 12, Page 437, Vol. 12, No. 2, pp. 437. DOI: 10.3390/ELECTRONICS12020437.
3. **Baldwin, B., Ross, A. (2001).** Baldwin language technology's DUC summarization system. Proceedings of the 1st Document Understanding Conference, New Orleans, LA.
4. **Baxendale, P. B. (1958).** Machine-Made Index for Technical Literature—An Experiment. IBM Journal of Research and Development, Vol. 2, No. 4, pp. 354–361. DOI: 10.1147/rd.24.0354.
5. **Belwal, R. C., Rai, S., Gupta, A. (2023).** Extractive text summarization using clustering-based topic modeling. Soft Computing, Vol. 27, No. 7, pp. 3965–3982. DOI: 10.1007/S00500-022-07534-6/METRICS.
6. **Boros, E., Kantor, P. B., Neu, D. J. (2001).** A Clustering Based Approach to Creating Multi-Document Summaries. Technical report.
7. **Dhaini, M., Erdogan, E., Bakshi, S., Kasneci, G. (2024).** Explainability meets text summarization: A survey. International Natural Language Generation Conference, pp. 631–645.
8. **Edmundson, H. P. (1969).** New Methods in Automatic Extracting. Journal of the ACM, Vol. 16, No. 2, pp. 264–285. DOI: 10.1145/321510.321519.
9. **El-Kassas, W. S., Salama, C. R., Rafea, A. A., Mohamed, H. K. (2021).** Automatic text summarization: A comprehensive survey. DOI: 10.1016/j.eswa.2020.113679.
10. **Fattah, M. A., Fattah, M. A. (2014).** A hybrid machine learning model for multi-document summarization. Appl Intell, Vol. 40, pp. 592–600. DOI: 10.1007/s10489-013-0490-0.
11. **Kostiuk, Y., Pichardo-Lagunas, O., Malandii, A., Sidorov, G. (2023).** Automatic detection of semantic primitives using optimization based on genetic algorithm. PeerJ Computer Science, Vol. 9, pp. e1282. DOI: 10.7717/PEERJ-CS.1282.
12. **Lin, C.-Y., Hovy, E.**, **.** NeATS in DUC 2002. Technical report.
13. **Louis, A., Nenkova, A. (2013).** Automatically Assessing Machine Summary Content Without a Gold Standard. Computational Linguistics, Vol. 39, No. 2, pp. 267–300. DOI: 10.1162/COLI_A_00123.
14. **Luhn, H. P. (1958).** The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development, Vol. 2, No. 2, pp. 159–165. DOI: 10.1147/rd.22.0159.
15. **Mahlow, C., Ulasik, M. A., Tuggener, D. (2024).** Extraction of transforming sequences and sentence histories from writing process data: a first step towards linguistic modeling of writing. Reading and Writing, Vol. 37, No. 2, pp. 443–482. DOI: 10.1007/s11145-021-10234-6.
16. **Mendoza, M., Cobos, C., León, E., Lozano, M., Rodríguez, F., Herrera-Viedma, E. (2014).** A New Memetic Algorithm

for Multi-document Summarization Based on CHC Algorithm and Greedy Search. Springer, Cham, pp. 125–138. DOI: 10.1007/978-3-319-13647-9_14.

17. **Neri-Mendoza, V., Ledeneva, Y., García-Hernández, R. A. (2020).** Unsupervised extractive multi-document text summarization using a genetic algorithm. Journal of Intelligent & Fuzzy Systems, Vol. 39, No. 2, pp. 2397–2408. DOI: 10.3233/JIFS-179900.
18. **Neri Mendoza, V., Ledeneva, Y., García-Hernández, R. A., Hernández Castañeda, Á. (2024).** Relevance of sentence features for multi-document text summarization using human-written reference summaries. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 14755 LNCS, pp. 319–330. DOI: 10.1007\textslash978-3-031-62836-8_30.
19. **Qaroush, A., Abu Farha, I., Ghanem, W., Washaha, M., Maali, E. (2021).** An efficient single document Arabic text summarization using a combination of statistical and semantic features. Journal of King Saud University - Computer and Information Sciences, Vol. 33, No. 6, pp. 677–692. DOI: 10.1016/J.JKSUCI.2019.03.010.
20. **Rezaei, H., Amid Moeinzadeh Mirhosseini, S., Shahgholian, A., Saraee, M., Shahgholian AShahgholian, A., Rezaei, H. (123).** Features in extractive supervised single-document summarization: case of Persian news. Language Resources and Evaluation. DOI: 10.1007/s10579-024-09739-7.
21. **Rojas-Simón, J., Ledeneva, Y., García-Hernández, R. A. (2018).** Calculating the significance of automatic extractive text summarization using a genetic algorithm. Journal of Intelligent and Fuzzy Systems, Vol. 35, No. 1, pp. 293–304. DOI: 10.3233/JIFS-169588.
22. **Rojas-Simón, J., Ledeneva, Y., García-Hernández, R. A. (2021).** Evaluation of text summaries without human references based on the linear optimization of content metrics using a genetic algorithm. Expert Systems with Applications, Vol. 167, pp. 113827. DOI: 10.1016/J.ESWA.2020.113827.
23. **Supriyono, Wibawa, A. P., Suyono, Kurniawan, F. (2024).** A survey of text summarization: Techniques, evaluation and challenges. Natural Language Processing Journal, Vol. 7, pp. 100070. DOI: 10.1016/J.NLP.2024.100070.
24. **Torres-Moreno Juan-Manuel (2014).** Cognitive Science and Knowledge Management Series. Automatic Text Summarization.
25. **Vázquez, E., García-Hernández, R. A., Ledeneva, Y. (2018).** Sentence features relevance for extractive text summarization using genetic algorithms. Journal of Intelligent & Fuzzy Systems, Vol. 35, No. 1, pp. 353–365. DOI: 10.3233/JIFS-169594.
26. **Verma, P., Om, H. (2019).** MCRMR: Maximum coverage and relevancy with minimal redundancy based multi-document summarization. Expert Systems with Applications, Vol. 120. DOI: 10.1016/j.eswa.2018.11.022.
27. **Verma, S., Vagisha Nidhi (2019).** Extractive Text Summarization Using Deep Learning. Proceedings - 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018, pp. 43–56. DOI: 10.1109/ICCUBEA.2018.8697465.
28. **Xiong, Y., Yan, M., Hu, X., Ren, C., Tian, H. (2023).** An unsupervised opinion summarization model fused joint attention and dictionary learning. Journal of Supercomputing, Vol. 79, No. 16, pp. 17759–17783. DOI: 10.1007/S11227-023-05316-X/METRICS.
29. **Yang, Y., Tan, Y., Min, J., Huang, Z. (2024).** Automatic text summarization for government news reports based on multiple features. Journal of Supercomputing, Vol. 80, No. 3, pp. 3212–3228. DOI: 10.1007/S11227-023-05599-0/METRICS.

*Article received on 23/03/2024; accepted on 02/06/2024.*
**Corresponding author is Yulia Ledeneva.*

ISSN 2007-9737

# Comparison of Performance of Amazon Braket Using a Quantum Genetic Algorithm

Sandra S. Rosales[1], Oscar Montiel[1], Ulises Orozco-Rosas[2], Juan J. Tapia[1], Oscar Castillo[3,*]

[1] Instituto Politécnico Nacional,
Centro de Investigación y Desarrollo de Tecnología Digital,
Mexico

[2] El Centro de Enseñanza Técnica y Superior,
Mexico

[3] Instituto Tecnológico de Tijuana,
Mexico

{oross, jtapiaa}@ipn.mx, srosales@citedi.mx,
ulises.orozco@cetys.mx, ocastillo@tectijuana.mx

**Abstract.** This study evaluates the performance of a Quantum Genetic Algorithm (QGA) executed on quantum devices. The algorithm was implemented in MATLAB and Python and deployed on the Amazon Web Services (AWS) platform. The QGA was utilized to optimize a set of continuous single-variable functions. The implementation employed the Hadamard quantum gate to initialize the population and the $R_y$ rotation gate for mutation and crossover. The findings revealed significant differences in execution time and costs were observed between the two implementations, undescoring the performance of quantum devices available on AWS. The results demonstrate that the QGA can achieve optimal solutions in a few generations, suggesting its potential for efficiently solving complex problems. However, the costs and availability of quantum devices remain restrictive. This work exemplifies the potential of leveraging AWS cloud-based quantum computing platforms for the research and development of quantum algorithms.

**Keywords.** Quantum computing, quantum genetic algorithms, mathematical optimization, quantum device on AWS, quantum metaheuristic algorithm.

## 1 Introduction

Quantum computing has experienced significant growth in both the development of devices and the application of quantum algorithms across various fields [8]. One prominent area is optimization, which can be approached using different methods, including evolutionary algorithms [25].

In the context of our work, optimization involves the process of maximizing or minimizing a continuous single-variable function. The concept of quantum evolutionary algorithms was first introduced by Narayanan and Moore in 1996 [19]. Their work laid the groundwork by applying the principles of quantum mechanics to the evolution of a quantum particle over time.

This idea has been enriched over the years, and various applications in quantum simulators have existed. For example, some studies such as [3, 10] applied this approach to combinatorial optimization, while [17] used it for optimization functions.

Different proposals for applying this method are generally discussed in works like [14, 18].

**Algorithm 1** Quantum genetic algorithm (QGA)

**Require:** g, $s_p$ ▷ g: amount of generations, $s_p$: population size
**Ensure:** The best solution **b**
**procedure** QGA(g, $s_p$)
  $t \leftarrow 0$
  Set up a quantum population $Q_t(s_p)$ with quantum gate $H$
  Measure each individual in $Q_t(s_p)$ to create a population $P_t(s_p)$
  Save the best solution **b** in $P_t(s_p)$
  **while** $t <$ g **do** ▷ $t$: indicate the number of generation
    **Update** $Q_t(s_p - 1)$ applying a quantum rotation gate $U(\Delta\theta_j)$
    Measure each individual in $Q_t(s_p - 1)$ to create a population $P_t(s_p - 1)$
    Save the best solution **b** in $P_t(s_p - 1)$
    $t \leftarrow t + 1$
  **end while**
**end procedure**

Quantum computing offers significant potential for solving problems that classical computing cannot efficiently address. However, it is important to note that current quantum devices are in the NISQ (Noisy Intermediate-Scale Quantum) era, meaning that their results may include errors.

These errors can be mitigated by repeating executions, which unfortunately makes the process slow and costly. Therefore, the problems addressed must be adapted to the current limitations of these quantum devices. As technology advances, it is anticipated that the need for repeated measurements will decrease.

Amazon Braket, a service from Amazon Web Services (AWS), has emerged as a valuable tool providing access to quantum devices from various vendors. This platform enables users to work with different types of quantum devices, as discussed in Section 3. Access to these quantum devices can be achieved through Python or MATLAB. In Section 3 we review the pseudocode of a Quantum Genetic Algorithm (QGA) that was coded and executed in MATLAB and Python.

This section details how to establish a connection to AWS, enabling code execution on quantum devices. Additionally, Section 4 presents the results of implementation using both languages. The goal of this study is to compare the performance of a QGA implemented to maximize a set of continuous single-variable functions using both Python and MATLAB.

The structure of this paper is as follows: Section 2 covers the main concepts and foundational knowledge employed in this study. Section 3 explores the implementation details, algorithms, functional aspects, and provides a brief cost analysis of using AWS for quantum computing. Section 4 describes the findings from applying this methodology with both a simulator and a quantum device. The final Section 5 includes the conclusions and suggests potential avenues for future research.

## 2 Theoretical Framework

This section presents a review of problem optimization and fundamentals of quantum computing that will aid in understanding Quantum Genetic Algorithms (QGAs).

### 2.1 Optimization Problem

Optimization problems are prevalent across various fields, including engineering, logistics, finance, and artificial intelligence. These problems involve finding the optimal solution, often within complex constraints or large solution spaces [20]. Traditional optimization techniques, such as linear programming or gradient descent, are effective for well-structured problems but frequently struggle with highly nonlinear, multi-modal, or combinatorial optimization challenges [6].

These conventional methods often encounter difficulties in identifying optimal solutions within a practical time frame. This is where metaheuristic methods become essential [4]. Metaheuristic methods encompass a class of optimization algorithms designed to tackle complex and computationally demanding optimization problems. Unlike deterministic approaches, metaheuristics rely on heuristic or rule-of-thumb strategies, avoiding explicit mathematical models [11]. These algorithms explore solution spaces by adeptly

**Table 1.** Values and conditions for rotation $\Delta\theta_i$[1]. $\mathbf{X_i}$ depict the $i-th$ decimal value proposed. $\mathbf{B_i}$ depict the $i-th$ decimal value stored

| $\mathbf{x_{ij}}$ | $\mathbf{b_{ij}}$ | $\mathbf{f(X_i) \geq f(B_i)}$ | $\mathbf{\Delta\theta_i}$ |
|---|---|---|---|
| 0 | 0 | $false$ | 0 |
| 0 | 0 | $true$ | 0 |
| 0 | 1 | $false$ | 0 |
| 0 | 1 | $true$ | $-0.05\pi$ |
| 1 | 0 | $false$ | $-0.01\pi$ |
| 1 | 0 | $true$ | $0.025\pi$ |
| 1 | 1 | $false$ | $0.005\pi$ |
| 1 | 1 | $true$ | $0.025\pi$ |

**Algorithm 2** Update of the QGA

**Require:** $q_i, x_i, b_i$ ▷
$q_i$ quantum individual, $x_i$ represents the best element, while $b_i$ represents the new candidate element in binary form. ▷ $f(\mathbf{B_i})$ and $f(\mathbf{X_i})$ where $\mathbf{B_i}$ and $\mathbf{X_i}$ are decimal values
**Ensure:** Rate of evolution
1: **procedure** UPDATE($q_i$)
2: $i \leftarrow 0$
3: **while** $i < m$ **do** ▷ $m$ is the length of $q_i$
4: Calculate $\Delta\theta_i$ and $s(\alpha'_i, \beta'_i)$ with to Table 1
5: Update $q_i$ by applying :
6: $[\alpha'_i\beta'_i]^T = U(\Delta\theta_i)[\alpha_i, \beta_i]^T$
7: $i \leftarrow i + 1$
8: **end while**
9: $q_i = q'_i$
10: **end procedure**

balancing exploration to discover new areas of the landscape and exploitation to refine promising solutions.

The strength of metaheuristics lies in their adaptability and versatility, as they can be customized for various problem types and constraints. Additionally, they excel in solving real-world challenges where exact solutions are often impractical due to computational limitations [11]. As technology advances and optimization problems become increasingly complex, the role of metaheuristics continues to expand, establishing them as indispensable tools for researchers and practitioners tackling real-world optimization issues.

Furthermore, quantum computing and recent proposals of quantum metaheuristics enhance the search process within the optimization landscape, opening new frontiers for addressing previously unsolvable problems [7].

The objective of optimization is to find the best possible solution, referred to as *feasible solutions*, which are measured using numerical functions, referred to as *objective functions* [15]. In the *feasible solution* set, the solution that yields the best objective function value is referred to as the *optimal solution* [5]. The formal definition is shown in:

Let $f : X \to \mathbb{R}$ be a function where $X \subseteq \mathbb{R}^n$ $n-$dimensional in the form $x = [x_1, \ldots, x_n]^T$ The aim is to:

$$\begin{array}{ll} f(x)_{\text{max}} & \text{or} \quad f(x)_{\text{min}}, \\ x \in X_{\text{max}} & x \in X_{\text{min}}, \end{array} \tag{1}$$

where $x_j, j = 1, \ldots, n$ is identified as a *decision variable*, $X$ refers to the *feasible region* and $f$ is the *objective function*.

### 2.2 Quantum Computing Fundamentals

In quantum computing, the basic unit of information is a quantum bit commonly called a *qubit*, which has a form $|\psi\rangle$[21]. Quantum computing mathematically represents qubits using Dirac notation, also referred to as bra-ket notation. For example, the ket $|i\rangle = \begin{bmatrix} 1 & i \end{bmatrix}^T$ where $i \in \mathbb{C}$, and the ket $|1\rangle = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$. A qubit is a linear combination of $|0\rangle$ and $|1\rangle$.

Bra and ket vectors are complex vectors within the Hilbert space, existing in a dual space. The notation $\langle\psi|$ indicates a bra, which is the complex conjugate transpose of the ket vector $|\psi\rangle$. Hence, $\langle\psi| = |\psi\rangle^\dagger$, where the symbol † signifies the complex conjugate transpose operation.

In quantum computing, computational bases are fundamental for representing and manipulating information. The most common computational basis consists of the states $|0\rangle$ and $|1\rangle$, which are

**Table 2.** Types of quantum computers: Available quantum devices on Amazon braket. Source: The table is from the chapter [22]

| Companies | Quantum device names | Quantum computing technologies |
|---|---|---|
| IonQ | Harmony, Aria-1 | Through the utilization of precise laser pulses, this system monitors ions that are confined in space to execute quantum gate operations and measurements. The systems are equipped with 11 and 25 qubits respectively. |
| Oxford Quantum Circuits (OQC) | Lucy, Aspen, M-3 | Its technology utilizes superconducting quantum processors, having 8 qubits. In contrast, Rigetti is composed of two chips with a total of 79 qubits. |
| QuEra Computing | Aquila | The principle behind this technology is to use lasers to arrange and excite neutral atoms into highly energetic states. Their quantum computer consists of 256 qubits operating in analog mode. Analog Hamiltonian Simulation is his paradigm. |

analogous to the binary bits "0" and "1" in classical computing. These states serve as the foundation for encoding and processing quantum information. The *superposition principle* in quantum mechanics permits a qubit to be in a linear combination of the two fundamental states, $|0\rangle$ and $|1\rangle$. This relationship is illustrated in Eq. 2:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \text{ with } \alpha, \beta \in \mathbb{C}. \tag{2}$$

In Eq. 2, $\alpha$ and $\beta$ are complex numbers representing the probability amplitudes. When measuring a qubit, the likelihood of finding the qubit in state $|0\rangle$ or $|1\rangle$ is given by the square of the absolute value of $\alpha$ or $\beta$, respectively.

The normalization condition requires that the sum of these squares equals one: $|\alpha|^2 + |\beta|^2 = 1$. Classical computers use registers to store bits, likewise, quantum computers register and store qubits. Despite this, a quantum qubit register and a classical bit register differ in important ways.

Quantum bits, unlike classical bits, are in a superposition of states. This enables them to represent both "0" and "1" simultaneously, enabling quantum computers to handle multiple possibilities simultaneously. Furthermore, qubits are capable of becoming *entangled*, enabling a profound link between their states, despite being physically separated, which facilitates more efficient calculations. Eq. 3 represents a quantum register of size $n$ mathematically. A tensorial product of two states (vectors) is represented by the symbol $\otimes$:

$$\begin{aligned} |\Psi\rangle &= |\psi_1\rangle \otimes ... \otimes |\psi_n\rangle = |101\rangle = |1\rangle \otimes |0\rangle \otimes |1\rangle\,, \\ &= \begin{pmatrix}0\\1\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}0\\0\\0\\0\\0\\1\\0\\0\end{pmatrix}. \end{aligned} \tag{3}$$

*Quantum gates* are fundamental in quantum computing. They function similarly to classical logic gates in traditional computing, enabling control over the state of one or more qubits. Thanks to quantum gates, quantum computers can carry out specific tasks much faster than classical computers, as quantum states can exist in superposition and be entangled [27].

Operations can be performed on individual qubits or on groups of qubits, known as quantum registers. Some one-qubit quantum gates originate from the Pauli set. For instance, the $X = \left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ gate, which induces rotations around the $X$-axis, and the $Y = \left[\begin{smallmatrix}0 & -i\\ i & 0\end{smallmatrix}\right]$ gate, which causes rotations around the $Y$-axis.

Another significant one-qubit gate frequently used to bring a qubit into a superposition state

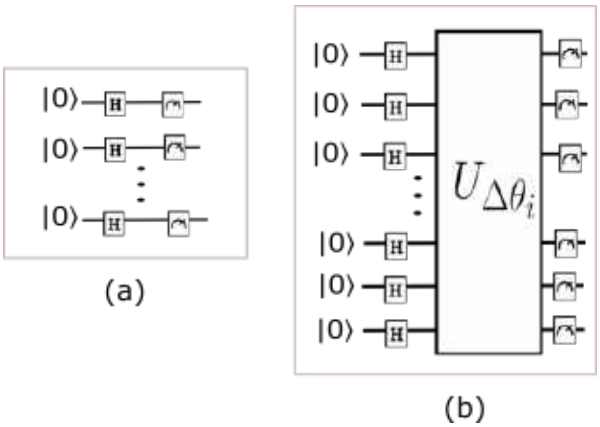



**Fig. 1.** The functions of the QGA are illustrated in the following diagram. (a) The quantum circuit for initializing the first population. (b) The quantum circuit for mutation and crossover operators. Figures taken from [22]

is the Hadamard gate, $H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$. This superposition creates equal probabilities for the qubit to be in the $|0\rangle$ or $|1\rangle$ states, with the $|1\rangle$ state having a relative phase difference of $\pi$ radians.

A one-qubit quantum gate acts on a quantum state $|\psi\rangle$ as $U|\psi\rangle$, where $U$ is a quantum operator such as the quantum gate $H$. For instance, applying the Hadamard gate $H$ to the quantum state $|0\rangle$ yields $H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 1\end{bmatrix}$, which simplifies to $H|0\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$. Another type of one-qubit gate is the rotation gate, which can be represented as:

$$R_x(\theta) = \begin{bmatrix}\cos\frac{\theta}{2} & -i\sin\frac{\theta}{2}\\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2}\end{bmatrix},$$
$$R_y(\theta) = \begin{bmatrix}\cos\frac{\theta}{2} & -\sin\frac{\theta}{2}\\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2}\end{bmatrix}, \quad (4)$$
$$R_z(\theta) = \begin{bmatrix}e^{-\frac{i\theta}{2}} & 0\\ 0 & e^{\frac{i\theta}{2}}\end{bmatrix}.$$

In Eq.4, $R_x(\theta)$ represents a rotation about the $x$-axis, $R_y(\theta)$ denotes a rotation about the $y$-axis, and $R_z(\theta)$ indicates a rotation around the $z$-axis. The variable $\theta$ stands for the angle of rotation for each respective axis. As mentioned earlier, a quantum register consists of multiple qubits grouped together as a single entity.

Similar to single quantum gates, in a quantum register, a quantum gate denoted as $U_r$ can operate on the quantum register state $|\psi_r\rangle^n$ of $n$ qubits in the form $U_r|\psi_r\rangle^n$. For example, the application of the $H^{\otimes 2}$ gate (which is two Hadamard gates in parallel, i.e., $H \otimes H$) on a two-qubit register $|\psi_r\rangle^2$ is represented as $H^{\otimes 2}|\psi_r\rangle^2$, which can be expressed as:

$$H^{\otimes 2}|\psi_r\rangle^2 = \frac{1}{2}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -1 & 1 & -1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}\begin{bmatrix}S_0\\ S_1\\ S_2\\ S_3\end{bmatrix}, \quad (5)$$
$$= \frac{1}{2}\begin{bmatrix}S'_0\\ S'_1\\ S'_2\\ S'_3\end{bmatrix}.$$

In this case, $S_0 \cdots S_3$ represent the quantum states of $|\psi_r\rangle^2$, and $S'_0 \cdots S'_3$ denote the quantum states that result from applying the quantum gate $H^{\otimes 2}$ to the state $|\psi_r\rangle^n$. In general, for a $n$-qubit state in the computational basis $|0\cdots 0\rangle_n$, applying the Hadamard gate to each qubit results in:

$$|\psi_r\rangle^n = \frac{1}{\sqrt{2^n}}(|0\cdots 000\rangle + |\cdots 001\rangle + \cdots + |1\cdots 111\rangle). \quad (6)$$

This operation with the Hadamard gate is crucial, as many quantum algorithms use it as an initial state. It enables all the $2^n$ orthogonal qubits in the basis states $|0\rangle$ and $|1\rangle$ to be placed into a superposition state with equal probabilities. In general, the procedure can be expressed as:

$$H^{\otimes n}|x\rangle^n = \frac{\sum_z (-1)^{x\odot z}|z\rangle}{\sqrt{2^n}}, \quad (7)$$

where $x \odot z$ denote the bit-by-bit inner product of $x$ and $z$.

Measurement plays a fundamental role in quantum mechanics, offering insights into physical observables and their associated probabilities. It is essential to understand that measuring a quantum system disturbs it, causing an irreversible change in its state. In quantum computing, measurement is

**Table 3.** Cost by quantum device per quantum task and shot. Information taken from Amazon braket pricing

| QPU family | Quantum task price (USD) | Per-shot price (USD)[2] |
|---|---|---|
| Harmony | $0.30 | $0.01 |
| Aria | $0.30 | $0.03 |
| Lucy | $0.30 | $0.00035 |
| Aspen-M3 | $0.30 | $0.00035 |

**Table 4.** Description of total quantum task, final shots, and final cost per experiment

| Experiment (shots) | Total quantum tasks | Total shots | Final cost (USD) |
|---|---|---|---|
| MATLAB (100) | 583 | 58,300 | $ 803 |
| Python (1,000) | 534 | 534,000 | $ 4,211 |
| | | **Total cost** | $ 5,014 |

particularly significant for retrieving the information encoded within the computational system. In the domain of quantum mechanics, a range of measurement models are utilized to illustrate the interaction between a quantum system and a measurement apparatus, as well as the acquisition of measurement outcomes [21].

A variety of measurement models are used such as the Projection Model, Expectation Value Model, State Collapse Model, Statistical Model, and Eigenvalue and Eigenspace Model.

The standard notation of measurements consists of a measurement operator represented as $M_m$, where $m$ is used as an index for a possible measurement outcome. If state is $|\psi\rangle$, the probability of getting a measurement result $m$ is as the following:

$$Pr(m) = \langle\psi| M_m^\dagger M_m |\psi\rangle . \tag{8}$$

Upon obtaining this measurement, the subsequent state will be:

$$|\psi'\rangle = \frac{M_m |\psi\rangle}{\sqrt{\langle\psi| M_m^\dagger M_m |\psi\rangle}}. \tag{9}$$

Quantum computing depends on measuring a group of qubits “in the computational basis”, which consists of the states $|0\rangle$ and $|1\rangle$. In this particular scenario, we consider the spin direction of individual qubits in the quantum memory register along the z-axis of the Bloch sphere, which traverses both the North and South poles of the sphere.

The results of these measurements indicate that each qubit is either aligned with the $z$-axis, with “spin-up”, or it is in opposition, with “spin-down”, which corresponds to state $|0\rangle$ and $|1\rangle$ respectively. If this measure is applied to each qubit in a quantum memory register comprising $n$-qubits, will be obtained one of $2^n$ possible bit strings configurations.

The generation of different outcomes is contingent upon the superposition of each binary string configuration present in the register immediately before measurement. To illustrate this, let us consider an $n$-qubit quantum memory register which is in the normalized state $\sum_{i=0}^{2^n-1} c_i |i\rangle$ (where $|i\rangle$ represents a bit-string).

There will be variability in the outcome depending on the magnitude of the amplitudes $c_i$ and whether it is making a full measurement (measuring all the qubits) or a partial measurement (measuring only a few qubits). The result will be $|i\rangle$, with a probability of $|c_i|^2$ for each state.

### 2.3 Quantum Genetic Algorithm

An evolutionary algorithm represents an optimization approach inspired by biological evolution, used to identify or approximate solutions to intricate problems spanning various disciplines [16]. These algorithms commence with a population of potential solutions, subjecting each to fitness evaluation via a predefined function.

Selection for reproduction is based on fitness, leading to the generation of new solutions through crossover and mutation. This iterative process unfolds across multiple generations [11]. Our work focuses on Quantum Genetic Algorithms (QGAs) [9], which are rooted in Quantum-Inspired Genetic Algorithms (QIGAs) [19]. The two types of algorithms are grounded in quantum principles but have distinct implementation and computational frameworks. Quantum-Inspired Genetic Algorithms (QIGAs) are classical

**Table 5.** Test functions

| Function | Global maximum $(x, f(x)),\ x \in [0, 127]$ |
|---|---|
| $f_1(x) = \frac{x}{10} \cdot \sin(\frac{x}{10})$ | (79.7867, 7.9167) |
| $f_2(x) = \frac{x}{2} + 5$ | (127.0, 68.5000) |
| $f_3(x) = -\frac{\cos(x) \cdot x^2}{10} + 3x$ | (122.5400, 1,868.9900) |
| $f_4(x) = \frac{(x - 135\pi)(\sin(\frac{x+135\pi}{10}))}{10}$ | (15.4535, 39.2826) |
| $f_5(x) = \frac{-x^2}{10} + 3x$ | (15.0800, 22.5000) |
| $f_6(x) = \begin{cases} \frac{\sin(x-20)}{x-20} & if \quad x \neq 20 \\ 0.9999 & if \quad x = 20 \end{cases}$ | (19.9946, 0.9999) |

**Table 6.** The best maximum found in two executions, one with 10 runs and the other with 40 runs, each consisting of 15 generations, was compared between *Harmony* and SV1 versus the local simulator on MATLAB

| Function $f_n$ | AWS Device MATLAB | Local Simulator MATLAB |
|---|---|---|
| $f_1$ | (80.0625, **7.9136**) | (81.9910, **7.9167**) |
| $f_2$ | (124.0000,67.0000) | (127.0000,**68.5000**) |
| $f_3$ | (116.1875,1696.7312) | (122.6571,**1,868.9434**) |
| $f_4$ | (15.4375, **39.2826**) | (18.6126, **39.2826**) |
| $f_5$ | (15.0000, **22.5000**) | (15.0762, **22.5000**) |
| $f_6$ | (20.0625, **0.9993**) | (19.9775, **0.9999**) |

algorithms that run on conventional computers. They are inspired by quantum principles and seek to replicate quantum effects using classical computing methods [23].

In contrast, QGAs are specifically created to operate on authentic quantum computers, exploiting the complete potential of quantum characteristics. They possess the capacity for substantial acceleration in solving certain problems, however, their practical utility is currently constrained by the early stage of quantum hardware development and limited accessibility.

The specific approach of this study is implementing a Quantum Genetic Algorithm (QGA) using two programming languages that allow connection to quantum computers via AWS. In MATLAB, the *'quantum'* package developed in 2023 was used. For Python, the *Amazon Braket* library was utilized. It is important to mention that these tools also allow simulations to be run on personal computers.

As discussed in [18], we have customized QGA for execution on quantum computers. We take the proposal developed in [22] where " $q_i$ characterizes a quantum system $|\psi_i\rangle$ that encompasses $2^m$ simultaneous states, as expressed in Eq. 10. Here, $m$ represents the genetic composition of each individual, and $i$ signifies the population size":

$$|\psi_i\rangle = \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{bmatrix} = q_i. \tag{10}$$

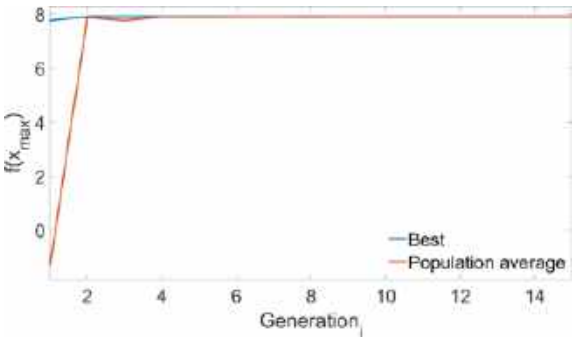
The outcome will produce a quantum population structured in the following manner:

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{bmatrix}_{q_1}, \\ \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{bmatrix}_{q_2}, \\ \vdots \\ \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{bmatrix}_{q_i}. \tag{11}$$

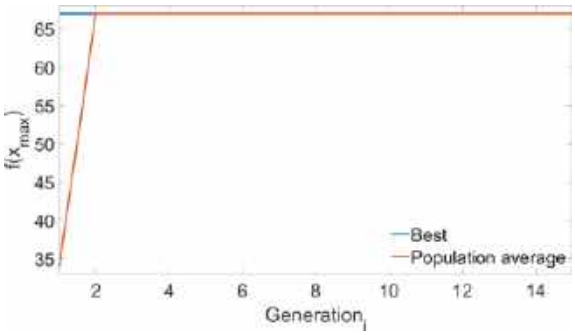
As a result, in [22], Algorithm 1 and Algorithm 2 were developed. The Algorithm 1 requires two elements: the total amount to the evolution of population, i.e., generations, g, and the population size $s_p \rightarrow i$. In more detail on line 2, the generation number $t$ is adjusted to $0$.

In line 3, we applied the quantum Hadamard gate to all the qubits, as defined in Eq. 7 to produce equal chance distributions for all potential individuals. In line 4, we obtain a classical population by measuring all individuals in the current population. In line 5, we store the best solution in the current population in "**b**".

The while-loop, spanning from lines 6 to 10, runs continuously as long as the generation counter remains less than the maximum number, indicated as "g". The process involves generating a new quantum population using the best solution "**b**" in line 7, followed by an update using Algorithm 2. A new classical population is then formed by measuring the individuals obtained in line 7 and line 8. In line 9, the best solution from the entire process is saved as "**b**". Additionally, the




**(a)**




**(b)**

**Fig. 2.** Plot of the best and average values as a function of generation. The algorithm converged in the early generations. (a) Function $f_1$ and (b) Function $f_2$ with $15$ generations on *Harmony*

generation counter is incremented at the end. The mutation is applied using the $R_y$ gate, which consists of two parameters: the qubit of the $i-th$ individual to which a certain rotation will be applied, and the degrees of rotation, assigned according to Table 1.

In [22] explained that " Algorithm 2 provides the required rotation angle that will eventually modify the amplitude probability to change individual qubits of the quantum chromosomes.

The algorithm relies on Table 1 to select the appropriate angle for its use in Eq. 12; this updates the corresponding qubit in the quantum chromosome". The original strategy proposed in [10] to rotate the angle was modified as shown in Table 1:

$$\left[\alpha'_x \beta'_x\right] = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \begin{bmatrix} \alpha_x \\ \beta_x \end{bmatrix}. \quad (12)$$

For example, the way that we adapted the quantum inspiration was to take only the $i-jth$ values and compare them, assigning a rotation based on said comparison, trying to ensure convergence towards a better individual. Suppose you have the best individual $b_i = 10111110011$ and the new individual obtained is $x_i = 10110110110$ plus their respective decimal values are $\mathbf{X_i}$ and $\mathbf{B_i}$. If we will be taken $x_{ij} = 1$ and $b_{ij} = 1$, with $i = 1$ and $j = 1$. $f(\mathbf{X_i})$ and $f(\mathbf{B_i})$ are compared, using the Table 1, if $f(\mathbf{X_i}) \geq f(\mathbf{B_i})$, it will assign $\Delta\theta_i = 0.025\pi$ otherwise $0.005\pi$.

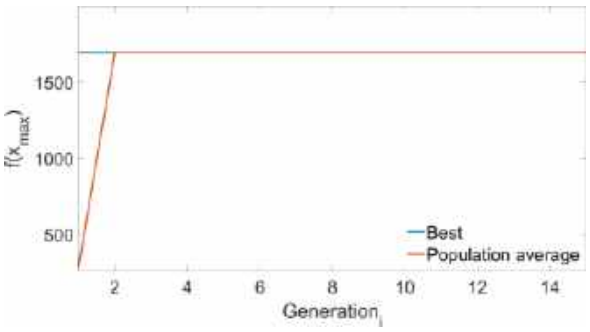
## 3 Platform Amazon Web Service

AWS platform is designed to connect companies and customers who use and develop software for various purposes. One development within the AWS platform is Amazon Braket which was designed for research and applications in the area of quantum computing. D-Wave was the first device available in the launch of this platform on August 12, 2020. At present, it offers access to many quantum hardware development companies. The Amazon Braket name derives from the quantum mechanics term "bra-ket".

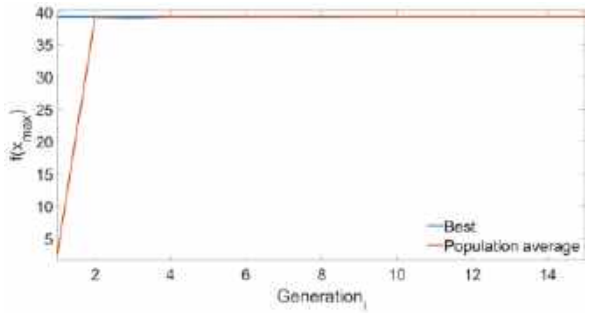
Currently, it provides access to quantum hardware developed by the companies listed in Table 2. By using this service, you can explore and design quantum algorithms such as Grover, Bernstein-Vazirani, and Deutsch-Jozsa, among others, execute them in different quantum simulators, run them on different quantum computers, and even demonstrate concepts about quantum computing.

Furthermore, it integrates Python notebook environments as well as MATLAB and PennyLane platforms. The way of working is in 3 stages: build, test, and execute. In addition, Amazon Braket enables the storage of all execution results and the running of algorithms without the requirement for individual configurations to specific device providers.

That is, it is enough to indicate the name of the quantum device that will be used. This makes it a convenient and easy-to-use platform for quantum computing. However, it is important to carefully review the available features of each




**(a)**




**(b)**

**Fig. 3.** Plot of the best and average values as a function of generation. The algorithm converged in the early generations. (a) Function $f_3$ on *Harmony* and (b) Function $f_4$ on *Simulator SV1* with $15$ generations, respectively

company, since some devices do not support certain quantum operations.

This is because the supplier companies constantly update their quantum devices, and it takes time for this information to reach AWS. The circuit model is currently used as the foundation for representing and executing quantum operations by several well-established universal quantum computing platforms.

Quantum operations are depicted as quantum gates in this model, and computations are arranged as a series of these gates. Some quantum computing platforms that implement this programming paradigm include “IBM Quantum Experience” [26], “Google’s Sycamore” [1], and “Rigetti’s quantum computing platform” [12]. However, with AWS it is possible to have access to different kinds of hardware quantum. In [22] it is detailed in Table 2 form some characteristics of quantum devices available with this platform.

Furthermore, it also explains the offered access to three simulators, SV1, DM1, and TN1, which use AWS resources. SV1 is a tensor network simulator with 50 qubits, DM1 is a vector state simulator with 34 qubits, and TN1 uses a density matrix with 17 qubits.

It is pertinent to mention that quantum gates work in the same manner regardless of the programming language used, whether it is MATLAB or Python. Qubits are manipulated by quantum gates through the use of pulses that are sent to them.

As a result of these pulses, a wave is generated that is triggered by a signal connected to an input port on the hardware. Different parameters make up a signal such as amplitude, frequency, and duration.

A quantum circuit is referred to as a quantum task. In this work, the quantum tasks represent the initial population operator, mutations, and crossovers. Each of these genetic operators is run for $n$ generations (iterations) per execution to determine which result is most likely to be obtained.

Furthermore, for each quantum task, a certain number of measurements (shots) must be taken to mitigate error. The required number of shots will vary depending on the chosen quantum device. This refers to obtaining the measurement.

Although, the number of shots can be customized to suit your needs and error mitigation. The Amazon website recommends a minimum of 2,500 shots per quantum task for the *Aria-1* device. In particular, the executions in MATLAB had 100 shots, while the experiment with Python had 1,000.

This number of shots was used to stay within the project budget. AWS is a versatile platform that has grown and adapted to the needs of developers and emerging technologies. This company provides a variety of services, including the management of large databases, with high security until their processing. In addition to the accessibility of high-performance computing, such as machine learning and quantum computing. However, it is important to review the costs involved in each of these tools.

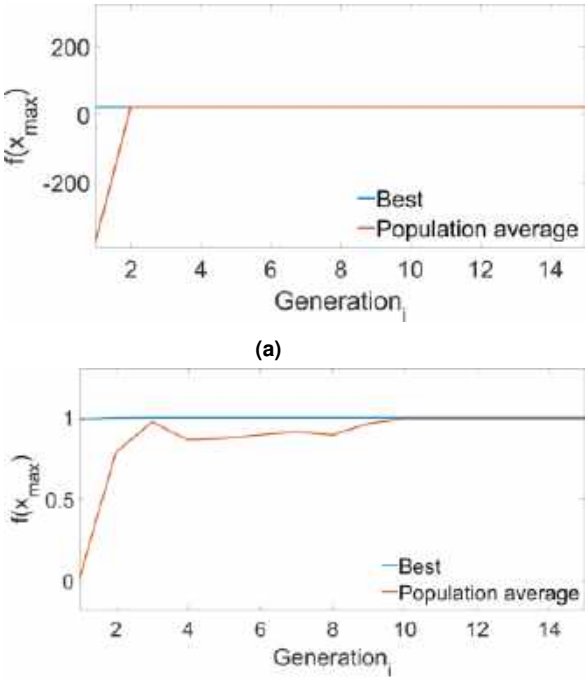



**Fig. 4.** Plot of the best and average values as a function of generation. The algorithm converged in the early generations. (a) Function $f_5$ and (b) Function $f_6$ with $15$ generations on simulator SV1

### 3.1 Implementation of QGA on Quantum Device in AWS

Referring to the design explained in Section 2.3, the resulting quantum circuits are depicted in Figure 1. For the first quantum population, we applied the *Hadamard* quantum gate described in Eq. 7. To achieve the quantum mutation and crossover processes, we used the $R_y(\theta)$ quantum gate described in Eq. 12.

The characteristics of QGA are taken from [22] where is used an array of 11 qubits to correspond to a single quantum chromosome. The population is constituted of 10 individuals.

A sequence of $m = 11$ qubits is defined as a quantum chromosome $q_i$, where $i \leq 10$. Therefore, we work with size chains of $11$, each containing $10$ elements. The precision was determined by the number of qubits available. In this case, 11 qubits were utilized. Among the devices listed in Table 2, only the *Harmony* from *IonQ* was accessible for MATLAB.

For Python, in addition to the *Harmony* device, *Aria-1* was also available. The domain was chosen for values of $x$ between $[0.00, 127.00]$. To ensure consistency in both implementations, a quantization method was employed to obtain decimal values, and they were represented with 4 decimal places after the decimal point.

The implementation of MATLAB with a specific quantum device is facilitated by the development of a library called *'quantum'*, which was introduced in 2023. The connection procedure is initiated through the command exemplified in Code3.1. It is noteworthy that the establishment of this connection necessitates the creation of an Amazon Web Services account. This applies to both Python and MATLAB, although the process differs for each.

```
loadenv('awsAccount.env')
region = "us-east-1";
bucketStoragePath = "s3://amazon-braket-name
    Bucket/nameFolder/";
deviceARN="arn:aws:braket:us-east-1::device/
    qpu/ionq/Harmony";
device = quantum.backend.QuantumDeviceAWS
    (devARN, Region=region, S3Path=
    bucketStoragePath);
% Quantum circuit example
qcExample = quantumCircuit(HGate(1));
taskExample = run(qcExample, device);
wait(taskExample);
measure = fetchOutput(taskExample);
% The next line shows results in a table from
table(measure.Counts, measure.MeasuredStates,
    VariableNames=["counts","States"])
```

One of the differences between MATLAB and Python, as demonstrated in Code 3.1, is that Python requires only a single line of code to set the session values and select the quantum device after installing the Amazon Braket library. An example is shown in Code 3.1.

```
#Required packages
!pip install amazon-braket-sdk
!pip install boto3
!pip install --upgrade amazon-braket-sdk
!pip install --upgrade amazon-braket-schemas

from boto3 import Session
from braket.aws import AwsDevice, AwsSession
from braket.circuits import Circuit
from braket.simulator import BraketSimulator
```

```
#Simulator
from braket.devices import LocalSimulator

# Use the awsAccount name _and_ region
session=Session(
aws_access_key_id='ID_awsAccount',
aws_secret_access_key='access_key_awsAccount',
region_name='us-east-1')
# Establish a Braket session with Boto3
aws_sessionQPU = AwsSession(boto_session
=session)
# Any QPU device with the previously
# initiated AwsSession should be instantiated.
device_arn =
'arn:aws:braket:us-east-1::device/qpu/ionq/
Aria-1'
device = AwsDevice(device_arn,
aws_session=aws_sessionQPU)
```

On the other hand, with MATLAB, it is necessary to declare different variables for correct functionality, as shown in Code 3.1 where the `device` variable represents the chosen quantum device, while the `taskExample` variable refers to the quantum task design (quantum circuit). For better clarity on the connection in MATLAB, the following steps are provided below:

1. Commence by obtaining and installing the *'quantum'* library. It is important to note that this library is compatible with MATLAB versions commencing from 2023a.
2. To facilitate daily task execution, create an AWS account of the IAM type. This process entails:
   (a) Authorizing another user and setting up an "access key" from the Summary section.
   (b) Giving the authorization for using Amazon Braket services.
   (c) Handling permissions for the cloud-based storage solution (bucket S3).
3. Document the login credentials in a text file with the format "filename.env". With the next structure:
   (a) `AWS_ACCESS_KEY_ID= **ACCESS IDENTIFIER GENERATED**`
   (b) `AWS_SECRET_ACCESSS_KEY= **PROVIDED ACCESS KEY**`
   (c) `AWS_DEFAULT_REGION= **REGION OF THE QUANTUM DEVICE**`
4. Start with the `loadenv` command to load the contents specified by the "filename.env".
5. Generate an instance of "Bucket" labeled so that it commences with `amazon-braket-`
6. Inside this instance (bucket), create a repository to keep the outcomes of the computations.
7. Extract "Copy S3 URI" located in the repository generated in the bucket. This will be our "bucketStoragePath".
8. Ensure that the region configured for the bucket and the quantum hardware are identical.
9. Look into the available AWS quantum devices and simulators.
10. Choose one quantum device and obtain it "Device ARN". This value should be written in Code 3.1 to the variable "deviceARN".
11. Implement the Code 3.1 to initiate the connection to MATLAB.

**Table 7.** The best element obtained only in $f_4$ and $f_6$ with 3 runs of 10 generations in *Aria-1* the others functions are marked with '-' and 40 runs with 15 generations in the Local simulator on Python

| Function $f_n$ | AWS Device Python | Local Simulator Python |
|---|---|---|
| $f_1$ | - | **7.9153** |
| $f_2$ | - | **68.4705** |
| $f_3$ | - | 1,856.0263 |
| $f_4$ | **39.2008** | **39.2819** |
| $f_5$ | - | **22.4994** |
| $f_6$ | **0.9998** | **0.9901** |

One essential component for storing the outcomes and requirements of our operations is the "Amazon S3 bucket" (Amazon Simple Storage Service). This service facilitates the storage of data as objects in a bucket, with a maximum storage capacity of 5 TB. It allows for the storage of various

**Table 8.** The mean, standard deviation, and median of the better elements obtained on 40 runs with 15 generations on the Local simulator on MATLAB and Python

| $f_n$ | Local Simulator MATLAB | | | Local Simulator Python | | |
|---|---|---|---|---|---|---|
| | Mean | SD | Median | Mean | SD | Median |
| $f_1$ | **7.9099** | 0.0259 | 7.9167 | **7.6636** | 0.7396 | 7.8710 |
| $f_2$ | **68.4592** | 0.1499 | 68.5000 | 66.9243 | 2.1947 | 67.6314 |
| $f_3$ | 1867.5001 | 3.8520 | 1868.9434 | 1486.0648 | 277.0820 | 1537.3369 |
| $f_4$ | **39.1616** | 0.4689 | 39.2826 | 38.3043 | 1.7572 | 39.1145 |
| $f_5$ | **22.4999** | 0.0001 | 22.5 | 21.6097 | 4.9062 | 22.3917 |
| $f_6$ | **0.9998** | 0.0002 | 0.9999 | **0.5754** | 0.3784 | 0.6355 |

file types, including videos, text files, and Braket task results.

In contrast, Python does not require manual configuration for generating necessary instances for storage; it automatically does so using the SageMaker tool. Therefore, the use of these tools should be periodically monitored.

The quantum task process is maintained in the same way in both languages because it is carried out internally by AWS. A task, which can be a quantum circuit or a quantum register, is defined and sent to the device for execution. In certain instances, the task is placed in a queue and held until the quantum device or simulator is prepared to receive it.

Following the submission of jobs to a quantum device, third-party companies with quantum computers process the tasks. Upon completion in MATLAB, the results are securely streamed to an "S3 bucket". Effective monitoring and management of all tasks can be performed on the "Quantum Tasks" page through the "Amazon Braket console".

### 3.2 Cost Analysis

Amazon Braket is currently an excellent option for diving into quantum computing, especially for countries with limited accessibility to different quantum hardware. However, it is important to consider the available budget to make the most of this tool. It is crucial to know the concept of a quantum algorithm for this reason.

In Section 2.2, it is explained that a quantum algorithm is a set of quantum gates that perform a specific task. However, we are in the NISQ era, where noise affects the consistency of results. This noise makes a single execution insufficient to guarantee accuracy due to the probabilistic nature of quantum outcomes.

Therefore, multiple measurements are required to statistically reduce the effect of variability caused by noise. In Amazon Braket, it is possible to configure the number of shots per quantum task according to specific needs, depending on the selected device. For example, with *Aria-1*, it recommends 2,500 shots.

In Table 3, we present the cost of each shot per quantum device and quantum tasks. It is crucial to emphasize that these devices are from multiple companies and utilize distinct technologies. Although our project did not focus on a specific technology, the type of technology used may be relevant for other types of problems. For instance, in our case, only the number of qubits available was relevant for a better representation of decimal numbers.

In our QGA process described in Subsection 2.3, we can observe two quantum tasks. The first task involves obtaining an initial quantum population using the Hadamard gate Eq. 7, while the second task involves executing mutation and crossover applied through by the quantum gate $R_y(\theta)$ Eq. 12, where $\theta$ is set using Table 1.

The final cost of each experiment is presented in Table 4. As detailed in the implementation section, each execution requires at least two other AWS tools, which incur additional costs for execution time, storage, and instance creation. Moreover, the cost varies based on the city chosen for tool execution.

Table 4 shows the costs derived from two experiments that consist of optimizing six different functions using MATLAB and Python. The executions in MATLAB had 100 shots, while the experiment with Python had 1,000. This number of shots was used to stay within the project budget. If the AWS recommendation of using 2,500 shots for more reliable measurements had been followed, the total cost for both experiments would have been USD$84,110.1, in contrast to the USD $5,014 spent. The costs associated with AWS encourage the search for different options to work

with quantum devices. One possible proposal is the purchase of a quantum device.

For example, purchasing a device "Gemini Mini"[3] with a price of around $5,000, which is equipped with 2 qubits and based on the theory of nuclear magnetic resonance (NMR) [24]. However, this option is limited compared to Amazon Braket due to the restricted number of available qubits and the reliance on a single technology.

Therefore, the choice of quantum hardware will depend on the research objectives and the available budget. In this sense, Amazon Braket offers a wider range of quantum technology options.

## 4 Experiments and Results

This section presents the experiments and results implementing the QGA on a single-variable optimization problem using two platforms available to connect with quantum devices on AWS. The first platform utilizes the *'quantum'* library for MATLAB, and the second uses the Python programming language through "Amazon Braket". Finally, results from local simulators enabled on both platforms are compared. The QGA is assessed on different test functions with different levels of complexity to evaluate the strengths and weaknesses of the QGA. Table 5 presents the selected test functions with their respective global maximum to be found.

The goal is to determine if there is a difference in performance, specifically regarding the time required, and to ascertain if there is a variance in the results obtained from the function optimization problem. The assumption is that since we employ the same algorithm for both platforms, any deficiency should be evident in both sets of results.

The objective of QGA is to determine the value of $x$ that maximizes the functions shown in Table 5. Individuals were initially encoded in binary format and then converted to decimal values, as described in subsection 3.1. Two experiments were conducted. The first experiment aimed to define and observe the requirements and behavior of the connection with the MATLAB platform, and the second with Python. In both experiments, the precision of the QGA was examined.

[3] spinquanta.com/products-solutions/gemini

### 4.1 Experiment 1: MATLAB Implementation

The experiment for the MATLAB implementation involved running fifteen generations per test, with 100 shots for each quantum circuit. The *Harmony* device with 11 qubits was used for the first three test functions, and the *Amazon SV1* simulator was used for the last three test functions.

These adjustments were made to stay within the project budget, and due to the variability of costs in available quantum devices shown in Table 3, the number of runs in this test was 10. For comparison with the local simulator, 40 executions with 15 generations were carried out.

From this experiment, an average of all the best results obtained was calculated, as well as their standard deviation, as shown in Table 8 in comparison subsection 4.3 between local simulators. In the case of the MATLAB experiment with a quantum device, only the final solution of the 10 runs was obtained. Table 6 displays the final results.

The first column presents the best tuple $(x, f(x))$ obtained using the *Harmony* quantum device, and the second column shows the best result obtained with the quantum simulator. The value close to the maximum is in bold. It is important to note that both used the 'quantum' package.

In table 6, a relevant feature is presented regarding the approximation of the value of $x$. Quantization, which depends on the number of available qubits, is used to enhance the approximation when converting bits to decimal numbers. With 11 qubits, it is evident that the decimal values of $x$ differ from the value indicated in Table 5. Nevertheless, in this experiment, this variance did not have a significant impact on the QGA objective.

Table 6 shows the MATLAB approximations performed on the *Harmony* device and Local Simulator. The executions on the Local Simulator were on an Intel Core i7-9750 Processor (2.6 GHz) with 16 GB of RAM running Windows 11 Home with MATLAB 2023.

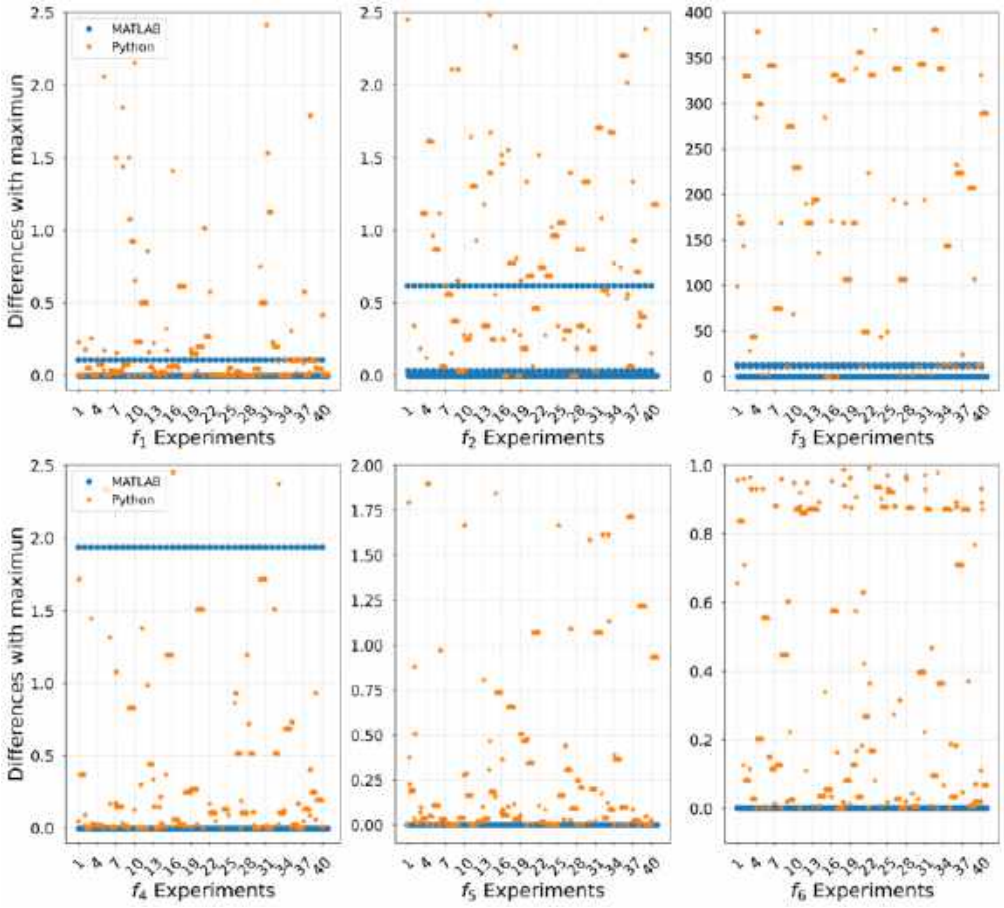



**Fig. 5.** Differences between the best elements found and the maximum of all functions

Regarding the behavior of the Quantum Genetic Algorithm (QGA) on MATLAB, the results depicted in Figures 2, 3, and 4 illustrate that the QGA requires a few generations to reach optimal or near-optimal values.

They also have an average that describes how the function values near the optimum or even the optimum value have been found since early generations. This is a promising outcome, especially compared to simulation results from previous studies such as [2], [13] and [17] which predict favorable results.

### 4.2 Experiment 2: Python Implementation

In the second experiment, we ran ten generations per test for the Python implementation of the QGA, using 1,000 shots on $f_4$ and $f_6$ with the *Aria-1* device, which contains 25 qubits. These limitations were set to stay within the budget. The number of executions was limited to three.

For the local simulator with the same characteristics mentioned in the previous experiment, we had 40 algorithm executions (runs). A comparison of the simulator run and the data obtained from the *Aria-1* quantum device run is shown in Table 7.

In Table 7, we can see that the results on the quantum device are near the maximum, while the local simulator also shows greater accuracy in the other functions. The closest values are highlighted in bold.

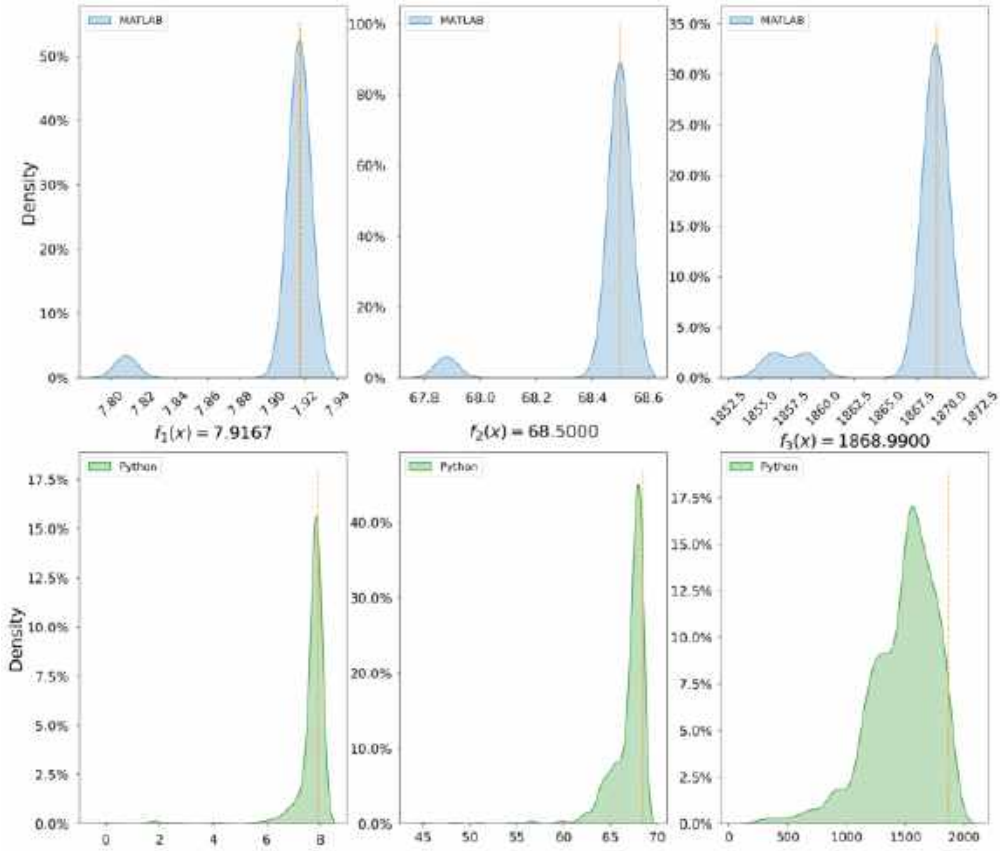



**Fig. 6.** Distribution of maximum findings for functions $f_1$ - $f_3$. Figures top MATLAB, bottom Python

### 4.3 Comparison of Local Simulator

Finally, we compare the two simulations made with the QGA versions. One simulation used the MATLAB *'quantum'* package and the other used *Amazon Braket* in Python with our computing resources (Intel Core i7-9750 Processor (2.6 GHz) with 16 GB of RAM running Windows 11 Home with MATLAB 2023 and Python 3.11) to simulate quantum processing. Table 8 shows the mean, standard deviation, and median.

The MATLAB data shows a lower standard deviation compared to Python, as well as a good mean approximation toward the target. In Table 8, it is possible to observe that the median results closest to the maximum for Python are with $f_1$ and $f_6$, which also have a lower deviation. In the case of MATLAB, $f_1, f_2, f_4, f_5$, and $f_6$ test functions show good performance, data that are shown in bold.

In Figure 5, the difference between the maximum value to be obtained, and the values generated by QGA on both platforms is shown where each function exhibits different behaviors. Additionally, the behavior with both libraries shows significant differences, which are consistent with Table 8. A high dispersion is observed for the case of Python (marked with a star).

The observed wiggly points are due to the random nature of the algorithm in every run, where each execution begins with a new quantum population that evolves after each generation (15). With MATLAB, the jumps are more noticeable as it approaches the last generation of each run, containing better elements and resulting in almost zero difference.

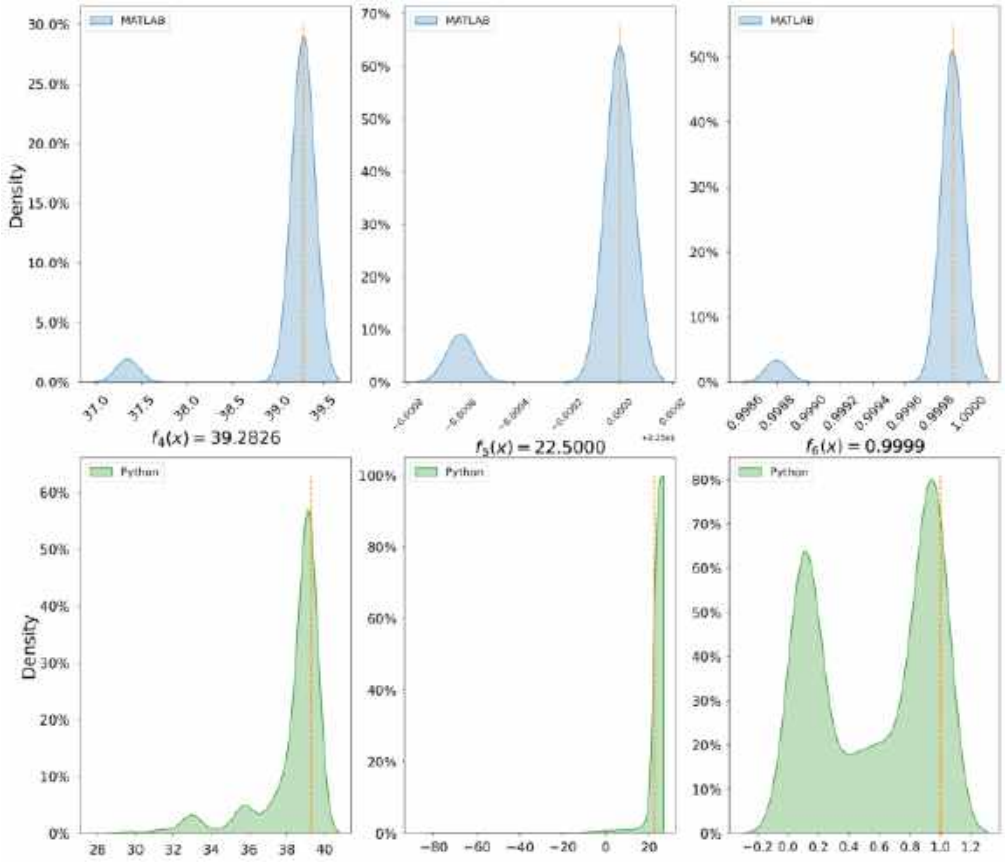



**Fig. 7.** Distribution of maximum findings for functions $f_4$ - $f_6$. Figures top MATLAB, bottom Python

In contrast, with Python, only some runs show improvement after each generation. For example, for test functions $f_1$, $f_4$, and $f_5$, a behavior similar to that of MATLAB can be seen, but not for the other functions.

The distribution of the total of the best individuals obtained from the local simulator executions is shown in Figure 6 and Figure 7. Results obtained with Python are represented in green, while those obtained with MATLAB are in blue, with a dotted line indicating the maximum required value.

Both Figures (6 and 7) demonstrate the precision of MATLAB. Although Python and MATLAB both show that the highest percentage of the values obtained are close to the maximum required, it is evident that Python produces results that are further away from the maximum required value compared to MATLAB.

On the other hand, Table 9 shows the execution times with MATLAB using *Harmony* and with Python using *Aria-1*, as well as the times with the local simulator, available on both platforms. Additionally, Table 9 shows the particular time to generate the quantum population and the quantum mutation (quantum tasks), data obtained from the record that AWS makes on the Amazon Braket console.

Additionally, it is possible to observe a significant difference in the time described in Table 9. The average time results shown in Table 9 were obtained from approximately 583 quantum tasks with 100 shots in MATLAB. For Python results, approximately 534 tasks with 1,000 shots. It is imperative to provide additional information

**Table 9.** Mean, standard deviation, and total time of the QGA execution with MATLAB and Python on a quantum device (3 and 10 runs with 15 and 10 generations respectively), and on a local simulator (40 runs with 15 generations each)

| | **MATLAB - *Harmony*** | | | **MATLAB - *Local Simulator*** | | |
|---|---|---|---|---|---|---|
| | Mean (s) | SD | **Total Time(s)** | Mean (s) | SD | **Total Time(s)** |
| **Quantum population** | 1,320 | 0.0206 | - | - | - | - |
| **Quantum mutation** | 2,100 | 0.0563 | - | - | - | - |
| Total mean | 2,040 | 0.0545 | - | 18.4259 | 9.5568 | - |
| - | - | - | 1,354.8 | - | - | 110.5556 |
| | **Python - *Aria-1*** | | | **Python - *Local Simulator*** | | |
| **Quantum population** | 36.5996 | 23.5761 | - | - | - | - |
| **Quantum mutation** | 36.5996 | 23.5761 | - | - | - | - |
| Total mean | 73.1993 | 47.1523 | - | 37.5076 | 0.0079 | - |
| - | - | - | 512.3955 | - | - | 225.0478 |

on operating hours to access specified quantum devices, such as *Harmony* and *Aria-1*.

These devices are available Monday through Friday from 12:00:00 to 03:00:00 UTC (Coordinated Universal Time). The above detail is significant since it introduces an additional waiting time to the total duration of the execution, a factor that is not considered in the current analysis because this time is independent of the chosen platform/programming language.

Likewise, AWS offers accessibility mechanisms such as *Braket Direct*, which facilitates time-bound requests for the use of specific quantum devices, and *Hybrid Jobs*, which encompasses hybrid quantum tasks.

The results shown in Figures 5, 6, and 7 depict the precision of the QGA in each test function. In that sense, statistics on the differences from the established maximum were obtained. A noticeable difference can be observed in test function $f_3$ due to its multimodal behavior.

However, in general, we can see that for the rest of the test functions, the range of differences is much closer to the global maximum. The behavior of the QGA for the optimization of single-variable multimodal functions shows results consistent with previous studies, such as those by [13, 17, 18].

These studies conducted on simulators predict a good performance of the QGA. One of the significant advantages is the reduced number of generations or individuals required to find the maximum, resulting in substantial time and computational resource savings.

This performance is observed even in devices that, to date, contain a limited number of qubits. Two notable aspects of this work are time and cost, as access to a quantum device depends on the stable connection with AWS and the connection between AWS and the company owning the quantum computer.

Additionally, the queue of jobs waiting for each device and their established response times are crucial factors. These factors are important to consider when working with AWS, as the availability of various devices may be affected.

For example, when attempting to execute a task on certain devices, such as *Aquila*, the device may be suspended by the company, resulting in execution errors. Additionally, internet availability is another crucial factor to consider.

## 5 Conclusion and Future Work

This study demonstrated the feasibility of implementing Quantum Genetic Algorithms (QGA) implemented via Amazon Braket on MATLAB and Python platforms. Moreover, it highlited relevant details, such as the dependence on the stability of the connection with AWS and the response times of the quantum computers' owner companies for accessing quantum devices.

The queue of jobs and internet availability also play crucial roles. These factors must be considered when working with AWS, as device availability may vary. For instance, certain devices might be suspended by their companies, leading to execution errors.

On the other hand, the QGA performance in optimizing single-variable multimodal functions is consistent with previous studies conducted on simulators. The QGA requires fewer generations or individuals to find the maximum, thus saving time and computational resources. This efficiency is maintained even on devices with a limited number of qubits. Quantum metaheuristics represent a promising advancement over classical metaheuristics, particularly for tackling complex high-dimensional problems.

Classical metaheuristics have demonstrated remarkable results, and the significant resources required emphasize the necessity of further exploration into quantum approaches. These approaches leverage fundamental quantum physics features such as superposition and entanglement, potentially leading to exponential improvements in execution time and resource efficiency.

Despite the challenges posed by the NISQ era, including quantum errors, noise, limited qubit availability, and hardware variability, this study provides a practical initial application of quantum devices, yielding the expected results.

Furthermore, the use of simulations accentuates the utility of widely adopted programming languages, such as Python and MATLAB, making quantum computing more accessible to the scientific community. This accessibility is crucial for advancing research and development in this rapidly evolving field.

Future work should explore multimodal optimization problems using different technologies to determine if certain technologies are more suitable for specific problems. This approach will help assess the comparative performance of various technologies.

For initial forays into quantum computing or when analyzing various quantum algorithms without a budget or clear time frame, cloud-based tools such as Qiskit and Pennylane are recommended.

Errors on the AWS platform can be costly in terms of money and time. AWS is best suited for developing well-established problems where exact execution costs can be calculated or when a large budget is available.

In conclusion, while the current state of quantum hardware presents several challenges, the continued development and expansion of quantum devices on platforms like Amazon Braket promise to enhance quantum computing capabilities.

Future research should focus on improving QGAs and exploring their applications across different technologies to address complex optimization problems more effectively.

## Acknowledgments

We thank the National Council of Humanities, Sciences, and Technologies (CONAHCYT) and Instituto Politécnico Nacional for supporting our research activities through project numbers CF-2023-I-108 and SIP20240164, respectively. This work was funded by the Red-CI Baja-AWS project of the Government of the State of Baja California.

## References

1. **AbuGhanem, M., Eleuch, H. (2023).** Experimental characterization of Google's sycamore quantum AI on an IBM's quantum computer. DOI: 10.2139/ssrn.4299338.

2. **Acampora, G., Chiatto, A., Vitiello, A. (2023).** Genetic algorithms as classical optimizer for the quantum approximate optimization algorithm. Applied Soft Computing, Vol. 142, pp. 110296. DOI: 10.1016/j.asoc.2023.110296.
3. **Ballinas, E., Montiel, O. (2023).** Hybrid quantum genetic algorithm with adaptive rotation angle for the 0-1 knapsack problem in the IBM Qiskit simulator. Soft Computing, Vol. 27, No. 18, pp. 13321–13346. DOI: 10.1007/s00500-022-07460-7.
4. **Blum, C., Roli, A. (2003).** Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys, Vol. 35, No. 3, pp. 268–308. DOI: 10.1145/937503.937505.
5. **Butenko, S., Pardalos, P. M. (2014).** Numerical methods and optimization: An introduction. CRC Press.
6. **De-los-Cobos, S. G., Goddard-Close, J., Gutiérrez-Andrade, M. A., Martínez-Licona, A. E. (2010).** Búsqueda y exploración estocástica. Universidad Autónoma Metropolitana.
7. **Dey, S., Bhattacharyya, S., Maulik, U. (2019).** Quantum inspired metaheuristics for image analysis.
8. **Gill, S. S., Kumar, A., Singh, H., Singh, M., Kaur, K., Usman, M., Buyya, R. (2022).** Quantum computing: A taxonomy, systematic review and future directions. Software: Practice and Experience, Vol. 52, No. 1, pp. 66–114. DOI: 10.1002/spe.3039.
9. **Han, K. H., Kim, J. H. (2000).** Genetic quantum algorithm and its application to combinatorial optimization problem. Proceedings of the 2000 Congress on Evolutionary Computation, Vol. 2, pp. 1354–1360. DOI: 10.1109/CEC.2000.870809.
10. **Han, K. H., Kim, J. H. (2002).** Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Transactions on Evolutionary Computation, Vol. 6, No. 6, pp. 580–593. DOI: 10.1109/TEVC.2002.804320.
11. **Haupt, R. L., Haupt, S. E. (2004).** Practical genetic algorithms. Wiley, Hoboken, New Jersey, USA.
12. **Karalekas, P. J., Tezak, N. A., Peterson, E. C., Ryan, C. A., da-Silva, M. P., Smith, R. S. (2020).** A quantum-classical cloud platform optimized for variational hybrid algorithms. Quantum Science and Technology, Vol. 5, No. 2, pp. 024003. DOI: 10.1088/2058-9565/ab7559.
13. **Lahoz-Beltra, R. (2016).** Quantum genetic algorithms for computer scientists. Computers, Vol. 5, No. 4, pp. 24. DOI: 10.3390/computers5040024.
14. **Malossini, A., Blanzieri, E., Calarco, T. (2008).** Quantum genetic optimization. IEEE transactions on evolutionary computation, Vol. 12, No. 2, pp. 231–241. DOI: 10.1109/TEVC.2007.905006.
15. **Marler, R. T., Arora, J. S. (2004).** Survey of multi-objective optimization methods for engineering. Structural and multidisciplinary optimization, Vol. 26, pp. 369–395. DOI: 10.1007/s00158-003-0368-6.
16. **Mitchell, M. (2001).** An introduction to genetic algorithms. Bradford, Cambridge, Massachusetts, USA.
17. **Montiel, O., Rubio, Y., Olvera, C., Rivera, A. (2019).** Quantum-inspired acromyrmex evolutionary algorithm. Scientific Reports, Vol. 9, No. 1. DOI: 10.1038/s41598-019-48409-5.
18. **Montiel-Ross, O. H. (2020).** A review of quantum-inspired metaheuristics: Going from classical computers to real quantum computers. IEEE Access, Vol. 8, pp. 814–838. DOI: 10.1109/ACCESS.2019.2962155.
19. **Narayanan, A., Moore, M. (1996).** Quantum-inspired genetic algorithms. pp. 61–66. DOI: 10.1109/ICEC.1996.542334.

20. **Quiroz-Castellanos, M., de-la-Fraga, L. G., Lara, A., Trujillo, L., Schütze, O. (2023).** Numerical and evolutionary optimization 2021. Mathematical and Computational Applications, Vol. 28, No. 3, pp. 71. DOI: 10.3390/mca28030071.
21. **Rieffel, E. G., Polak, W. H. (2011).** Quantum computing: A gentle introduction.
22. **Rosales-Alvarado, S. S., Montiel, O., Orozco-Rosas, U., Tapia, J. J. (2024).** Developing a quantum genetic algorithm in MATLAB using a quantum device on AWS. Springer Nature Switzerland, Cham, pp. 111–127. DOI: 10.1007/978-3-031-53713-4_10.
23. **Rubio, Y., Olvera, C., Montiel, O. (2021).** Quantum-inspired evolutionary algorithms on IBM quantum experience. Engineering Letters, Vol. 29, No. 4, pp. 1573–1584.
24. **Sels, D., Dashti, H., Mora, S., Demler, O., Demler, E. (2020).** Quantum approximate Bayesian computation for NMR model inference. Nature machine intelligence, Vol. 2, No. 7, pp. 396–402.
25. **Sun, Y., Xiong, H. (2014).** Function optimization based on quantum genetic algorithm. Research Journal of Applied Sciences, Engineering & Technology, Vol. 7, No. 1, pp. 144–149.
26. **Wootton, J. R., Harkins, F., Bronn, N. T., Vazquez, A. C., Phan, A., Asfaw, A. T. (2021).** Teaching quantum computing with an interactive textbook. 2021 IEEE International Conference on Quantum Computing and Engineering (QCE), pp. 385–391. DOI: 10.1109/QCE52317.2021.00058.
27. **Yanofsky, N. S., Mannucci, M. A. (2009).** Quantum computing for computer scientists.

*Article received on 23/05/2024; accepted on 18/07/2024.*
**Corresponding author is Oscar Castillo.*

ISSN 2007-9737

# Stop-Word Lists in Keyphrase Extraction: Their Influence and Comparison

Svetlana Popova[1], Mikhail Alexandrov[2,3,*], John Cardiff[1]

[1] Technological University Dublin,
Ireland

[2] Russian Academy of National Economy and Public Administration, Moscow,
Russia

[3] FRUCT Association,
Finland

svp@list.ru, malexandrov@mail.ru, john.cardiff@it-tallaght.ie

**Abstract.** Keyphrases provide a compact representation of a document's content and are useful in Web search systems, text data mining, and natural language processing applications. The keyphrase extraction domain has been developing for a long time, and achieving further improvements is becoming increasingly challenging. Algorithms compete for minimal gains, highlighting the significance of demonstrating ways to enhance the quality of both existing algorithms and those yet to be developed. This article aims to demonstrate and approve a simple way to enhance keyphrase extraction algorithms by using extended stop words. This enables the improvement of keyphrase extraction algorithms on average by 4% and more.Nevertheless, no studies have been conducted that compare different stop-word lists and their impact on the domain. Our goal is to overcome this gap. We compared the impact of both existing extended and standard stop-word lists on the performance of 10 unsupervised keyphrase extraction algorithms across 5 datasets (a total of 10 sub-datasets were used). We aimed to highlight that researching methods for constructing and using extended stop-word lists deserves attention and could become one of the sub-directions in the keyphrase extraction domain. Extended stop words, when a suitable list is selected, consistently enhance the performance of algorithms in a stable and statistically significant manner. Based on the obtained results, we can assume that knowing the type of text from which keyphrases need to be extracted allows us to select the most appropriate stop-word list.

**Keywords.** Keyphrase extraction, stop words, NLP.

## 1 Introduction and Related Work

Keyphrases succinctly summarize a document's content and the process of automatic keyphrase extraction involves the automated identification of significant and topic-relevant phrases from a text. Keyphrases play an important role in enhancing the capabilities of information retrieval systems [15, 3, 37, 6, 12] and contribute to natural language processing applications, e.g. document clustering [16], text classification [20], opinion mining [2], text summarization [21, 9], web tagging [27], and more, making the extraction of keyphrases an important area of data mining. Typically, keyphrases range from one to five words in length. The example of text (scientific abstract) with its keyphrases is presented in Table 1.

This study is dedicated to exploring unsupervised methods based on keyphrase candidate extraction. These methods include two steps: initially, potential keyphrases are identified and extracted from the text; subsequently, these candidates are scored and ranked to determine the final keyphrases. Unsupervised keyphrase extraction techniques can be categorized into

**Table 1.** Example: scientific abstract and keyphrases from the INSPEC dataset

| |
|---|
| **E-government**.The author provides an introduction to the main issues surrounding E-government **modernisation** and **electronic delivery** of all **public services** by 2005. The author makes it clear that E-government is about transformation, not computers and hints at the special **legal issues** which may arise. |

graph-based [29, 35, 14, 7, 5], statistical-based [31, 13, 8] and embedding/transformer-based [1, 23, 25, 10, 24, 34, 11, 33] approaches.

The results in the field are still far from high, and algorithms compete with each other for minimal improvements. Therefore, even small improvements play a role when comparing algorithms that extract keyphrases. In [36], the work of [26] is cited as the work where authors remove some words that are too common to be keywords. In our paper we call this kind of words: extended stop words. The list of these words from [26] is not publicly available [36].

[36] reports the 5% drop in performance of [26] approach without this list. It indicates that the influence of incorporating extended stop-words can be tangible. These 5% allow the algorithm from [26] to outperform the results of the algorithm from [36]. Without removing the specified words, the approach from [36] performs better than [26]. However, only a few studies specifically RAKE [31] and our study [30] focus on extended stop words in the keyphrase extraction domain and propose approaches to automate the process of extracting these words from the texts.

The authors of YAKE [8] also use a feature that helps reduce the weight of common words that do not reflect the context. The Word Relatedness to Context feature in YAKE looks like an attempt to find words similar to stop words in a hidden form. There are no studies in the field that compare different extended stop-word lists or examine their impact on the performance of existing algorithms.

To address this issue, we compared different stop-word lists (common and extended) on 10 unsupervised KE algorithms and 10 subsets of five datasets. Obtained results demonstrate that exploiting different stop-word lists affects the quality with which algorithms process test collections, and extended stop-word lists can noticeably improve the evaluation quality of algorithms.

An additional aim of this work is to highlight that researching methods for constructing and using extended stop-word lists deserves attention and could be one of the sub-directions in the KE domain.

Extended stop words enhance the performance of algorithms. At the end of the paper, we will show that improvement is achieved in most cases when extended stop-word lists are used with an average improvement of 4.5%-6% (in some cases, such improvement reaches up to 16%).

All studies in this work were conducted using datasets consisting of abstracts of scientific papers. Keyphrase extraction from these types of texts has attracted attention due to the active development of electronic libraries and e-learning platforms. Experiments were performed based on the Python-based Keyphrase Extraction framework (PKE) [4], which guarantees their correctness and reproducibility.

## 2 Keyphrase Extraction Problem, PKE Settings, Evaluation, and Datasets

### 2.1 Keyphrase Extraction

Keyphrase Extraction is defined as follows. Let $D = \{d_1,\ d_2,\ \ldots,\ d_n\}$ be a set of $n$ documents. Each document $d_i \in D$ has reference phrases (also called gold standard) – a set of keyphrases predefined by the experts $C_i = \{c_{i_1},\ c_{i_2},\ \ldots,\ c_{i_m}\}$. The goal of an unsupervised keyphrase extraction approach is for each text $d_i \in D$ automatically extract a list of keyphrase candidates, score them, create a ranked list, and select $k$ (@$k$) top-ranked phrases as keyphrases $G_i = \{g_{i_1},\ g_{i_2},\ \ldots,\ g_{i_k}\}$ that should match the set of reference phrases as precisely as possible in terms of exact match Precision, Recall and $F1_{score}$ - F1 (described below in subsection 2.3 Evaluation).

### 2.2 PKE and Algorithms Settings

To guarantee the correctness of the implementation of the algorithms exploited in the research and the reproducibility we used PKE framework. PKE [4], a comprehensive Python-based Keyphrase Extraction framework, incorporates implementations of keyphrase extraction techniques that were state-of-the-art at the time of its creation.

To operate identically with all the unsupervised keyphrase extraction algorithms involved in the study we exploited all of them in the same way: as candidate-based approaches similar to how it was done in PKE paper [4]. We exploited the following unsupervised methods implemented in PKE:

Graph-based (TextRank, SingleRank, TopicRank, PositionRank, TopicalPageRank, MultipartiteRank) and statistics-based (FirstPhrase, TfIdf, KP-Miner, YAKE) [4].

In all experiments, candidate phrases are extracted from the texts as continuous sequences of nouns and adjectives that are not stop words and satisfy the following pattern: $< \mathrm{ADJ} >^* < \mathrm{NOUN} | \mathrm{PROPN} >+$. Consequently, all exploited methods differ only in the ranking step.

### 2.3 Evaluation

#### 2.3.1 Keyphrase Extraction Evaluation

PKE evaluates an algorithm using the exact match macro-average $\mathrm{F1}_{\mathrm{score}}$@k, "@k" means that only $k$ keyphrases can be extracted for each text. $\mathrm{F1}_{\mathrm{score}}$ comparing a set of automatically extracted keyphrases for the text with a set of keyphrases marked for the same text by experts (reference keyphrases).

The score for each text $d_i$ is calculated by comparing the set of keyphrases extracted for the text ($G_i$) with the set of reference phrases from that text ($C_i$). $\mathrm{Precision}_{d_i}$ and $\mathrm{Recall}_{d_i}$ are calculated, based on which the $\mathrm{F1}_{\mathrm{score}_{d_i}}$ is determined as follows:

$$\mathrm{Precision}_{d_i} = \frac{|(C_i \cap G_i)|}{|G_i|}, \quad (1)$$

$$\mathrm{Recall}_{d_i} = \frac{|(C_i \cap G_i)|}{|C_i|}, \quad (2)$$

$$\mathrm{F1}_{\mathrm{score}_{d_i}} = \frac{2 \times \mathrm{Precision}_{d_i} \times \mathrm{Recall}_{d_i}}{\mathrm{Precision}_{d_i} + \mathrm{Recall}_{d_i}}, \quad (3)$$

where $|C_i \cap G_i|$ - is the number of correctly extracted phrases for text $d_i$, $|G_i|$ - is the number of phrases automatically extracted by the algorithm from the text, $|C_i|$ - is the number of reference phrases for text $d_i$.

We use $k$=10 as the most frequent way of evaluation in the domain. We do not remove phrases that do not occur in the corresponding text from reference keyphrases. This practice exists in the domain, e.g. in [19, 35] and it makes Recall and $\mathrm{F}_{\mathrm{score}}$ higher. Following standard practice in the domain, we performed stemming on all phrases before evaluating their quality.

PKE uses an exact match $\mathrm{F1}_{\mathrm{score}}$: An automatically extracted keyphrase is considered a true positive if the reference phrases contain the same phrase. If there is a semantically equivalent but visually distinct phrase, it is considered a false positive. This is an evaluation error, one of the errors described in [17] that causes low-performance quality in the domain.

Despite this, the $\mathrm{F1}_{\mathrm{score}}$ is exploited in most papers to compare keyphrase extraction algorithms and is the main and standard evaluation approach in the domain. Throughout the text, the quality of the performance of keyphrase extraction algorithms will be understood as their $\mathrm{F1}_{\mathrm{score}}$ evaluation on test datasets.

#### 2.3.2 Statistical Tests

We use statistical tests to demonstrate statistically significant differences in the performance of 10 unsupervised keyphrase extraction algorithms when they exploit different stop-word lists. We compare the results of the algorithms that exploited two different stop-word lists pairwise. The Wilcoxon signed-rank test is used.

By the 'better quality of a stop-word list' or 'more suitable list', we mean the following. Consider two stop-word lists - list $A$ and list $B$. If, when exploiting list $A$, the keyphrase extraction algorithm performs better in terms of evaluation in $\mathrm{F1}_{\mathrm{score}}$ compared to the same algorithm using stop words from list $B$, then we assert that stop-words list $A$ has better quality and is more suitable.

**Table 2.** Datasets description: ”k.p. per text” = average number of keyphrases per text in the subset, ”pr.” = present (indicate what % of keyphrases from references appear in the document), ch. = characters

| Datasets | num. of doc. | doc. descriptions | assigned by | k.p. per text | pr. % |
|---|---|---|---|---|---|
| **INSPEC** | domains: Computers and Control and Information Technology | | | | |
| train | 1,000 | title+abstract | reader | 9.79 | 78.00 |
| validation | 500 | title+abstract | reader | 9.15 | 77.96 |
| test | 500 | title+abstract | reader | 9.83 | 78.70 |
| **SemEval (TA)** | domains: Distributed Systems, Information Searchand Retrieval, Distributed Artificial Intelligence - Multiagent Systems, Socialand Behavioral Sciences - Economics | | | | |
| train | 144 | title+abstract | reader+author | 15.44 | 42.16 |
| test | 100 | title+abstract | reader+author | 14.66 | 40.11 |
| **kp20k** | domains: Computer Science | | | | |
| test | 20,000 | title+abstract | author | 5.28 | 58.40 |
| validation | 20,000 | title+abstract | author | 5.27 | 58.20 |
| **PubMed test dataset** | domain: Biomedical | | | | |
| first 500 doc. | 500 | title+first 1,200 ch. | author | 5.40 | 84.54 |
| last 500 doc | 500 | title+first 1,200 ch. | author | 5.40 | 84.54 |
| **KPBiomed test dataset** | domains: Biomedical | | | | |
| first 2,000 doc. | 2,000 | title+abstract | author | 5.22 | 66.59 |
| second 2,000 doc. | 2,000 | title+abstract | author | 5.22 | 66.59 |

### 2.4 Datasets

All test collections contain short texts and are from a scientific domain. We rely on PKE built-in well-known datasets. All collections are taken from a single repository that PKE works with [1]. These include:

- **INSPEC** [2] [19]: as test collections we use ”test” and ”validation” subsets. There are 500 scientific publication titles with abstracts in each subset with uncontr (reader) manually assigned reference keyphrases.
- **SemEval2010** [3] [22] dataset with 100 full texts in the ”test” subset and 144 full texts in the ”train” subset. Both subsets for each text have combined manual author- and reader-assigned keyphrases as reference keyphrases. We exploited the SemEval2010 dataset in the following format. SemEval2010(TA) includes only titles and abstracts of articles, making it similar to the INSPEC collection.
- **kp20k** [4] [28]: ”test” and ”validation” subsets were used for evaluation. Each subset includes 20,000 abstracts with titles from scientific articles with author-assigned keyphrases.
- **Pubmed** [5] [32]: dataset contains 1,320 articles with full text and author-assigned keyphrases. Titles are separated from full texts but abstracts are not. For each text, we took the title and the first 1,200 characters of the full text, assuming that in this way we would be able to use most of the abstracts. We created two subsets: the first

[1] huggingface.co/taln-ls2n
[2] huggingface.co/datasets/taln-ls2n/inspec
[3] huggingface.co/datasets/taln-ls2n/semeval-2010-pre
[4] huggingface.co/datasets/taln-ls2n/kp20k
[5] huggingface.co/datasets/taln-ls2n/pubmed

subset includes the first 500 documents from the database and the the second subset consists of the last 500 texts from the database. These subsets have no intersection.

– **KPBiomed**[6] [18]: ”test” subset of this dataset includes 20,000 abstracts and titles with author-assigned keyphrases. We created two subsets from ”test”: the first subset includes the first 2,000 texts from the database and the the second subset consists of the second 2,000 documents from the database. These subsets have no intersection. We chose this subset size because processing a collection of 20,000 documents across all experiments takes quite a long time.

The first three collections contain texts primarily from Computer Science, while the latter two contain texts from the domain of Biomedicine. From the same repository where these collections are available, we took statistics, which are combined into a Table 2.

## 3 Experiment Description and Results

In this section, we will compare how different stop-word lists affect the quality of the keyphrase extraction algorithms. Standard and extended stop-word lists will be used. Before we move on to the description of the experiment, consider the methods for building extended stop-word lists.

### 3.1 Extended Stop Word Lists Extraction

There are only two algorithms in RAKE [31] and in our research [30] for automatic extraction words that are too common to be a part of a keyphrase. In both articles, these words act as delimiters between phrases. In experiments, we exploited these words in the same way as stop words and we call them: extended stop words. Both algorithms [31] and [30] in original papers extract additional stop words (phrase delimiters) using the same source: INSPEC ”train” documents set [19]. Therefore, the extended stop word lists obtained by each approach differ only due to the differences between the methods used to extract these lists.

[6] huggingface.co/datasets/taln-ls2n/kpbiomed

#### 3.1.1 Phrase Delimiters Extraction in RAKE

RAKE is one of the most rapid algorithms. The authors proposed a method for extracting words that act as phrase delimiters. RAKE uses them together with other phrase delimiters, e.g. punctuation or common stop words, to split the longest sequences of continuous words that are extracted as candidates.

A phrase delimiters list is created based on the INSPEC train dataset [19]: the set of documents with labeled keyphrases. The method picks words with a document frequency higher than a threshold $df>10$ that occur more frequently as adjacent to keyphrases than within them and appends these words to the delimiters list. The obtained delimiters list merged with standard stop words we call **”RAKE-PD”**.

The obtained list of delimiters improves RAKE’s performance on the INSPEC test dataset compared to using the Fox stop words [31]. Examples of words from this list: ”the, and, of, a, in, is, for, to, we, this, are, with, as, on, it, an, that, which, by, using, can, paper, from, be, based, has, was, have, or, at, such, also, but, results, proposed, show, new, these, used, however, our, were, when, one, not, two, study, present, ...”(this list is the first part from example in the original paper [31]).

#### 3.1.2 Extended Stop-word List Extraction: Alternative Approach

We suggested another way to extract extended stop words for keyphrase extraction [30]. Keyphrases are extracted as longest sequences of contiguous nouns and adjectives split at phrase delimiters: punctuation, extended, and common stop-word positions. There is no ranking step in [30]. Here the extended stop-word list is built based on a set of documents annotated with keyphrases (based on the INSPEC train dataset similar to RAKE).

The approach iterates over a set of nouns and adjectives in the training dataset vocabulary. It measures the $F1_{score}$ increase in performance produced by the keyphrase extraction algorithm on INSPEC train if this current word is considered as a stop word.

**Table 3.** Evaluation of keyphrase extraction approaches exploited different stop-words lists in terms of $F1_{score}@10$. "SW list" = Stop-Word list name. The best results are highlighted in bold. The second-highest results are marked in italics with *. ESW = ExtentedSW list, RAKE = RAKE-PD list

| SW list− > | NLTK | ESW | RAKE | Fox | Smart | NLTK | ESW | RAKE | Fox | Smart |
|---|---|---|---|---|---|---|---|---|---|---|
| **INSPEC** | | | **test** | | | | | **validation** | | |
| FirstPhr. | 28.48 | **30.09** | 28.66 | 28.40 | * *28.75* | 28.75 | 29.37 | 26.70 | 28.25 | * *28.78* |
| TextR. | 34.78 | **37.26** | 35.64 | 35.39 | * *36.15* | 33.60 | **35.12** | 32.62 | 33.49 | * *34.68* |
| SingleR. | 34.77 | **36.51** | 35.23 | 35.03 | * *35.66* | 33.90 | **34.92** | 32.44 | 33.65 | * *34.56* |
| TopicR. | 28.43 | **29.56** | 27.65 | 28.45 | * *28.52* | 27.78 | **28.51** | 25.90 | 27.47 | * *27.91* |
| Multipar.R. | 29.34 | **30.47** | 28.71 | * *29.38* | 29.37 | 28.82 | **29.64** | 27.35 | 28.76 | * *29.20* |
| PositionR. | 33.48 | **34.83** | 33.38 | 33.47 | * *34.05* | 33.09 | **33.82** | 31.30 | 32.87 | * *33.66* |
| TopicalP.R. | 34.44 | **36.28** | 35.14 | 34.88 | * *35.31* | 33.54 | * *34.15* | 31.88 | 33.43 | **34.32** |
| Tf-Idf | 35.46 | **36.29** | 34.36 | 35.24 | * *35.72* | 33.71 | * *34.22* | 31.99 | 33.80 | **34.41** |
| KP-Miner | 33.81 | **35.07** | 33.99 | 34.52 | * *34.94* | 32.62 | * *33.49* | 31.10 | 32.90 | **33.61** |
| YAKE | 35.08 | **36.00** | 34.06 | 34.78 | * *35.49* | 33.60 | **33.96** | 31.32 | 32.98 | * *33.82* |
| **SemEval2010(TA)** | | | **test** | | | | | **train** | | |
| FirstPhr. | 15.37 | **16.61** | * *15.46* | 15.94 | 15.40 | * *17.11* | **17.74** | 16.47 | 16.63 | 16.56 |
| TextR. | 13.95 | * *15.84* | **16.23** | 14.99 | 15.13 | 16.01 | **17.58** | * *17.20* | 16.25 | 15.95 |
| SingleR. | 17.38 | **18.40** | * *17.93* | 17.80 | 18.30 | 18.00 | **19.31** | * *18.68* | 18.35 | 17.99 |
| TopicR. | 14.79 | **15.06** | 14.35 | * *14.91* | 14.81 | 16.40 | **17.04** | 15.97 | * *16.45* | 15.95 |
| Multipar.R. | 15.38 | **16.06** | 14.82 | * *15.95* | 15.35 | * *17.27* | **18.24** | 16.51 | 17.03 | 16.88 |
| PositionR. | 17.58 | **18.22** | 17.04 | 17.80 | * *18.00* | 18.94 | **20.29** | * *19.81* | 19.11 | 18.40 |
| TopicalP.R. | 16.82 | **18.10** | * *17.28* | 17.84 | 17.83 | 18.26 | **19.56** | * *18.61* | 18.54 | 18.17 |
| Tf-Idf | 16.35 | **16.83** | 15.82 | * *16.64* | 16.27 | * *18.61* | **19.12** | 18.12 | 18.12 | 17.79 |
| KP-Miner | 17.22 | **17.88** | * *17.54* | 17.59 | 17.53 | 18.56 | **19.63** | 18.53 | 18.40 | * *18.66* |
| YAKE | 18.64 | * *18.55* | 17.10 | 18.43 | 18.38 | 19.63 | * *19.49* | 18.76 | 19.46 | 19.33 |
| **kp20k** | | | **test** | | | | | **validation** | | |
| FirstPhr. | 13.50 | **13.99** | 13.42 | * *13.66* | 13.53 | 13.58 | **14.13** | 13.55 | * *13.74* | 13.63 |
| TextR. | 10.01 | **10.95** | **10.95** | 10.60 | 10.49 | 10.18 | **11.11** | * *11.05* | 10.85 | 10.75 |
| SingleR. | 12.52 | **13.16** | 12.91 | * *13.00* | 12.92 | 12.64 | **13.31** | 13.01 | * *13.17* | 13.06 |
| TopicR. | 11.97 | **12.35** | 11.91 | * *12.17* | 12.06 | 12.00 | **12.41** | 11.92 | * *12.19* | 12.09 |
| Multipar.R. | 13.55 | **13.95** | 13.40 | * *13.77* | 13.66 | 13.60 | **14.02** | 13.42 | * *13.78* | 13.66 |
| PositionR. | 14.08 | **14.57** | 14.17 | * *14.38* | 14.33 | 14.10 | **14.65** | 14.19 | * *14.45* | 14.37 |
| TopicalP.R. | 12.80 | **13.40** | 13.18 | * *13.28* | 13.19 | 12.95 | **13.58** | 13.27 | * *13.44* | 13.33 |
| Tf-Idf | 12.14 | **12.64** | * *12.52* | 12.49 | 12.46 | 12.27 | **12.80** | * *12.59* | 12.60 | 12.59 |
| KP-Miner | 14.05 | **14.46** | 14.03 | * *14.30* | 14.29 | 14.26 | **14.66** | 14.11 | * *14.51* | 14.46 |
| YAKE | 14.68 | **15.08** | 14.60 | * *14.88* | 14.81 | 14.75 | **15.22** | 14.64 | * *14.95* | 14.89 |

If a given improvement exceeds the threshold $h$ ($h$=0.0001), the word is labeled as a stop word. After every word is iterated, all words labeled as stop words are added to the final extended stop-word list as well as standard stop words from NLTK.

This final stop-words list we call **"ExtendedSW"**. Examples of words from this list: "entire, results, various, extensions, input, main, many, number, different, way, available, large, certain, ..." (this list is the first part from the original paper [30]).

**Table 4.** Evaluation of keyphrase extraction approaches exploited different stop-words lists in terms of $F1_{score}@10$. "SW list" = Stop-Word list name. The best results are highlighted in bold. The second-highest results are marked with an underscore. ESW = ExtentedSW list, RAKE = RAKE-PD list

| SW list$->$ | NLTK | ESW | RAKE | Fox | Smart | NLTK | ESW | RAKE | Fox | Smart |
|---|---|---|---|---|---|---|---|---|---|---|
| **Pubmed** | | | **first 500 doc.** | | | | | **last 500 doc.** | | |
| FirstPhr. | 14.71 | * *15.45* | **16.39** | 14.83 | 14.89 | 15.74 | * *16.19* | **16.55** | 15.66 | 15.83 |
| TextR. | 7.55 | * *8.20* | **8.68** | 8.03 | 7.98 | 8.20 | 8.47 | **8.92** | 8.38 | * *8.55* |
| SingleR. | 11.73 | * *12.40* | **12.71** | 11.96 | 11.87 | 12.37 | * *13.25* | **13.45** | 12.74 | 12.78 |
| TopicR. | 14.04 | 14.02 | **14.74** | * *14.31* | 14.21 | 14.37 | **14.52** | 14.43 | 14.29 | * *14.44* |
| Multipar.R. | 15.81 | * *16.15* | **17.07** | 16.09 | 15.98 | 16.32 | **16.84** | * *16.75* | 16.28 | 16.24 |
| PositionR. | 14.57 | * *15.18* | **15.88** | 14.84 | 14.81 | 15.12 | * *15.70* | **16.09** | 15.31 | 15.45 |
| TopicalP.R. | 12.15 | * *12.74* | **13.21** | 12.61 | 12.59 | 12.77 | * *13.45* | **13.98** | 13.07 | 13.17 |
| Tf-Idf | 15.92 | * *16.38* | **16.83** | 16.11 | 16.16 | 16.34 | * *16.59* | **16.93** | 16.30 | 16.58 |
| KP-Miner | 16.49 | * *16.59* | **16.82** | 16.38 | 16.48 | 16.97 | 17.05 | **17.31** | 17.04 | * *17.15* |
| YAKE | 16.05 | * *16.61* | **17.42** | 16.09 | 16.35 | 16.46 | * *16.96* | **17.22** | 16.60 | 16.75 |
| **KPBiomed** | | | **first 2000 doc.** | | | | | **second 2000 doc.** | | |
| FirstPhr. | 15.72 | * *16.26* | **16.42** | 15.74 | 15.92 | 15.60 | * *16.22* | **16.26** | 15.63 | 15.69 |
| TextR. | 6.91 | * *7.57* | **7.92** | 7.54 | 7.26 | 6.92 | * *7.63* | **7.84** | 7.32 | 7.30 |
| SingleR. | 10.95 | * *11.44* | **11.77** | 11.43 | 11.29 | 11.15 | * *11.91* | **12.21** | 11.63 | 11.56 |
| TopicR. | 13.49 | * *13.94* | **14.06** | 13.80 | 13.76 | 13.41 | * *13.72* | **13.75** | 13.62 | 13.61 |
| Multipar.R. | 15.71 | * *16.10* | **16.28** | 15.93 | 15.86 | 15.77 | * *16.19* | **16.22** | 16.00 | 15.91 |
| PositionR. | 13.84 | * *14.38* | **14.66** | 14.22 | 14.10 | 14.29 | * *14.80* | **15.11** | 14.61 | 14.59 |
| TopicalP.R. | 11.10 | * *11.73* | **12.09** | 11.53 | 11.43 | 11.23 | * *12.15* | **12.16** | 11.79 | 11.71 |
| Tf-Idf | 15.83 | * *16.06* | **16.24** | 16.05 | 16.00 | 16.08 | 16.31 | **16.44** | * *16.37* | 16.35 |
| KP-Miner | 16.68 | * *16.95* | **17.09** | 16.79 | 16.75 | 16.70 | 16.86 | **17.00** | * *16.91* | 16.87 |
| YAKE | 15.88 | * *16.34* | **16.54** | 16.18 | 16.10 | 16.19 | * *16.69* | **16.73** | 16.51 | 16.42 |

### 3.2 Different Stop-word Lists Comparison

#### 3.2.1 Experiment Description

We examined the impact of various stop-word lists on the performance of keyphrase extraction methods. We took the standard stop-word list from the NLTK, as well as the FOX[7] and SMART[8] stoplists previously tested in the domain of keyphrase extraction, and extended stop-word lists referenced in [31] RAKE (RAKE-PD) and in [30] (ExtendedSW). Each stop-word list was exploited in the work of each of the 10 unsupervised keyphrase extraction algorithms: TextRank, SingleRank, TopicRank, PositionRank, TopicalPageRank, MultipartiteRank, FirstPhrase, TfIdf, KP-Miner, and YAKE. Table 3 and Table 4 presents the results for each dataset. We conducted the Wilcoxon signed-rank tests to check whether some specific stop-word lists statistically significantly improved the quality of the algorithms compared with exploiting other stop-word lists.

[7] github.com/aneesha/RAKE/blob/master/FoxStoplist.txt

[8] github.com/aneesha/RAKE/blob/master/SmartStoplist.txt

#### 3.2.2 Results and Discussion

The results presented in Table 3 and Table 4 allow us to draw the following conclusions.

– On all five datasets, the extended stop word lists help the algorithms achieve the best results (there are only several exceptions). On the first three datasets, the best algorithm performance is achieved with the ExtendedSW list (ESW).

On the remaining two datasets, the RAKE-PD list outperforms ExtendedSW; however, ExtendedSW consistently remains the second most effective for these collections.

– The ExtendedSW stop word list allows algorithms to achieve the highest results for datasets related to the field of Computer Science, with only a few exceptions across all experiments. In the case of datasets from the Biomedical Sciences, the ExtendedSW list almost always yields the second-best results compared to the RAKE-PD stop word list.

  In other words, the ExtendedSW list consistently shows the best or second-best quality in nearly all experiments (except for 5 cases out of 100). The RAKE-PD stop word list enables algorithms to achieve the highest results on the two datasets from the field of Biomedicine, but in most cases, on three Computer Science datasets, this stop word list performs worse than the SMART or FOX lists.

  Therefore, we assume that the ExtendedSW list generally performs better than the RAKE-PD. Additionally, note that RAKE is a patented algorithm.

– The results obtained for stop-word lists on different subsets of the same datasets are closely similar. We can assume that an optimal stop word list for a given type of text can be selected using a subset of such texts for which reference keyphrases are available.

– On average, across all combinations of datasets and algorithms, ExtendedSW improves the performance of keyphrase extraction algorithms by 4% compared to the commonly used NLTK stop-word list. When considering only the datasets related to Computer Science, this improvement is 4.5%.

  ExtendedSW improves the performance of the algorithms in 98 out of 100 experiments. Compared to NLTK, the RAKE-PD list improved the performance of keyphrase extraction algorithms by 6% on Biomedical datasets. However, in more than half of the experiments on the datasets from Computer Science, exploiting RAKE-PD did not improve keyphrase extraction. Here, RAKE-PD falls behind ExtendedSW.

We conducted statistical tests to demonstrate that the results obtained on each Computer Science dataset using the list ExtendedSW statistically significantly improved algorithms' results achieved with the other stop-words lists: NLTK, FOX, SMART, and RAKE-PD. The Wilcoxon signed-rank test was used. The statistical test revealed statistically significant differences at the p-values 0.01 or 0.05 in all cases except one.

It is the comparison case with SMART list on the INSPEC "validation" subset where p-value=0.08. All other cases indicate that using the list ExtendedSW statistically significantly improved algorithms' results obtained with the other stop-words lists. The same for the RAKE-DP stop words on Biomedical datasets.

## 4 Conclusions

This work aimed to compare different stop-word lists and their impact on the keyphrase extraction domain. We compared standard and extended stop-word lists. We want to highlight that researching methods for constructing and using extended stop-word lists deserves attention. Experiments with 10 different unsupervised keyphrase extraction algorithms on 10 subsets from 5 different datasets show that extended stop-word lists allow the algorithms to achieve the best performance.

Obtained results show that the stop-word lists that allow keyphrase extraction algorithms to achieve the highest performance are very similar across different subsets of the same datasets. Additionally, we observed that the choice of a stop word list depends on the domain. For all datasets related to Computer Science, the best algorithm performance was achieved using the same extended stop-word list.

For Biomedical datasets, a different extended stop-word list proved to be the most suitable, but it was the same list across all Biomedical datasets. We assume that if we know the type of texts from which keyphrases need to be extracted, we can select the most appropriate stop-word list. On the domain-specific

datasets used in this study, extended stop-word lists enabled keyphrase extraction algorithms to achieve maximum performance, improving their quality by an average of 4.5% to 6%, with some algorithms showing up to a 16% improvement compared to using the standard NLTK stop-word list. These improvements justify the development of approaches for the automatic extraction of extended stop-word lists for keyphrase extraction tasks. The results also allow us to assume that the extended stop word list ExtendedSW is a good alternative to the extended stop words (phrase delimiters) extracted in the patented RAKE algorithm.

## References

1. **Bennani-Smires, K., Musat, C., Hossmann, A., Baeriswyl, M., Jaggi, M. (2018).** Simple unsupervised keyphrase extraction using sentence embeddings. DOI: 10.48550 /arXiv.1801.04470.
2. **Berend, G. (2011).** Opinion expression mining by exploiting keyphrase extraction. Proceedings of the 5th International Joint Conference on Natural Language Processing, pp. 11621170.
3. **Bernardini, A., Carpineto, C., D'Amico, M. (2009).** Full-subtopic retrieval with keyphrase-based search results clustering. Proceedings of the IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, Vol. 1, pp. 206–213. DOI: 10.1109/WI-IAT.2009.37.
4. **Boudin, F. (2016).** pke: An open source python-based keyphrase extraction toolkit. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, Osaka, Japan, pp. 69–73.
5. **Boudin, F. (2018).** Unsupervised keyphrase extraction with multipartite graphs. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Vol. 2, pp. 667–672. DOI: 10.18653/v1/N18-2105.
6. **Boudin, F., Gallina, Y., Aizawa, A. (2020).** Keyphrase generation for scientific document retrieval. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, pp. 1118–1126. DOI: 10.1 8653/v1/2020.acl-main.105.
7. **Bougouin, A., Boudin, F., Daille, B. (2013).** TopicRank: Graph-based topic ranking for keyphrase extraction. Proceedings of the Sixth International Joint Conference on Natural Language Processing, Asian Federation of Natural Language Processing, Nagoya, Japan, pp. 543–551.
8. **Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., Jatowt, A. (2020).** Yake! keyword extraction from single documents using multiple local features. Information Sciences, Vol. 509, pp. 257–289. DOI: 10.1016/j.ins.2019.09.013.
9. **D'Avanzo, E., Magnini, B. (2005).** A keyphrase-based approach to summarization: The lake system at DUC-2005. Proceedings of DUC.
10. **Ding, H., Luo, X. (2021).** Attentionrank: Unsupervised keyphrase extraction using self and cross attentions. Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1919–1928. DOI: 10.18653/v1/2021.emnlp-main.146.
11. **Ding, H., Luo, X. (2022).** AGRank: Augmented graph-based unsupervised keyphrase extraction. Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, Association for Computational Linguistics, pp. 230–239.
12. **Du, H., Thudumu, S., Giardina, A., Vasa, R., Mouzakis, K., Jiang, L., Chisholm, J., Bista, S. (2023).** Contextual topic discovery using unsupervised keyphrase extraction and

hierarchical semantic graph model. Journal of Big Data, Vol. 10, No. 1. DOI: 10.1186/s40537-023-00833-1.

13. **El-Beltagy, S. R., Rafea, A. A. (2010).** Kp-miner: Participation in SemEval-2. Proceedings of the 5th international workshop on semantic evaluation, pp. 190–193.
14. **Florescu, C., Caragea, C. (2017).** PositionRank: An unsupervised approach to keyphrase extraction from scholarly documents. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Vancouver, Canada, pp. 1105–1115. DOI: 10.18653/v1/P17-1102.
15. **Gutwin, C., Paynter, G., Witten, I., Nevill-Manning, C., Frank, E. (1999).** Improving browsing in digital libraries with keyphrase indexes. Decision Support Systems, Vol. 27, pp. 81–104. DOI: 10.1016/S0167-9236(99)00038-X.
16. **Hammouda, K. M., Matute, D. N., Kamel, M. S. (2005).** Corephrase: Keyphrase extraction for document clustering. Proceedings of the 4th International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer-Verlag, pp. 265–274. DOI: 10.1007/11510888_26.
17. **Hasan, K. S., Ng, V. (2014).** Automatic keyphrase extraction: A survey of the state of the art. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Baltimore, Maryland, pp. 1262–1273. DOI: 10.3115/v1/P14-1119.
18. **Houbre, M., Boudin, F., Daille, B. (2022).** A large-scale dataset for biomedical keyphrase generation. Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI), Association for Computational Linguistics, pp. 47–53. DOI: 10.18653/v1/2022.louhi-1.6.
19. **Hulth, A. (2003).** Improved automatic keyword extraction given more linguistic knowledge. Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, pp. 216–223. DOI: 10.3115/1119355.1119383.
20. **Hulth, A., Megyesi, B. B. (2006).** A study on automatically extracted keywords in text categorization. Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, USA, pp. 537–544. DOI: 10.3115/1220175.1220243.
21. **Jones, S., Lundy, S., Paynter, G. W. (2002).** Interactive document summarization using automatically extracted keyphrases. Proceedings of the 35th Annual Hawaii International Conference on System Sciences, pp. 1160–1169. DOI: 10.1109/HICSS.2002.994038.
22. **Kim, S. N., Medelyan, O., Kan, M. Y., Baldwin, T. (2010).** SemEval-2010 task 5: Automatic keyphrase extraction from scientific articles. Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics, pp. 21–26.
23. **Le, Q., Mikolov, T. (2014).** Distributed representations of sentences and documents. Proceedings of the 31st International Conference on Machine Learning, Vol. 32, No. 2, pp. 1188–1196. DOI: 10.48550/arXiv.1405.4053.
24. **Li, T., Hu, L., Sun, C., Li, S., Chi, L. (2021).** Triplerank: An unsupervised keyphrase extraction algorithm. Knowledge-Based Systems, Vol. 219, pp. 106846. DOI: 10.1016/j.knosys.2021.106846.
25. **Liang, X., Wu, S., Li, M., Li, Z. (2021).** Unsupervised keyphrase extraction by jointly modeling local and global context. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana,

Dominican Republic, pp. 155–164. DOI: 10.18653/v1/2021.emnlp-main.14.

26. **Liu, Z., Huang, W., Zheng, Y., Sun, M. (2010).** Automatic keyphrase extraction via topic decomposition. Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 366–376.
27. **Medelyan, O., Frank, E., Witten, I. H. (2009).** Human-competitive tagging using automatic keyphrase extraction. Association for Computational Linguistics, pp. 1318–1327.
28. **Meng, R., Zhao, S., Han, S., He, D., Brusilovsky, P., Chi, Y. (2017).** Deep keyphrase generation. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 582–592. DOI: 10.18653/v1/P17-1054.
29. **Mihalcea, R., Tarau, P. (2004).** TextRank: Bringing order into texts. Proceedings of the Empirical Methods in Natural Language Processing.
30. **Popova, S., Kovriguina, L., Mouromtsev, D., Khodyrev, I. (2013).** Stop-words in keyphrase extraction problem. Proceedings of the Conference of Open Innovation Association, FRUCT, pp. 113–121. DOI: 10.1109/FRUCT.2013.6737953.
31. **Rose, S., Engel, D., Cramer, N., Cowley, W. (2010).** Automatic keyword extraction from individual documents. Text Mining: Applications and Theory, pp. 1–20. DOI: 10.1002/9780470689646.ch1.
32. **Schutz, A. (2008).** Keyphrase extraction from single documents in the open domain exploiting linguistic and statistical methods. M. App. Sc Thesis.
33. **Song, M., Xu, P., Feng, Y., Liu, H., Jing, L. (2023).** Mitigating over-generation for unsupervised keyphrase extraction with heterogeneous centrality detection. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, pp. 16349–16359. DOI: 10.18653/v1/2023.emnlp-main.1017.
34. **Sun, Y., Qiu, H., Zheng, Y., Wang, Z., Zhang, C. (2020).** SIFRank: A new baseline for unsupervised keyphrase extraction based on pre-trained language model. IEEE Access, Vol. 8, pp. 10896–10906. DOI: 10.1109/ACCESS.2020.2965087.
35. **Wan, X., Xiao, J. (2008).** Single document keyphrase extraction using neighborhood knowledge. Proceedings of the 23rd National Conference on Artificial Intelligence, AAAI Press, Vol. 2, pp. 855–860.
36. **Wang, R., Liu, W., Mcdonald, C. (2015).** Using word embeddings to enhance keyword identification for scientific publications. pp. 257–268. DOI: 10.1007/978-3-319-19548-3_21.
37. **Zeng, H. J., He, Q. C., Chen, Z., Ma, W. Y., Ma, J. (2004).** Learning to cluster web search results. Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 210–217. DOI: 10.1145/1008992.1009030.

*Article received on 22/05/2024; accepted on 07/08/2024.*
**Corresponding author is Mikhail Alexandrov.*

ISSN 2007-9737

# Implementation of Best Practices for the Management of Radio-Electronic Signal Blocking Systems: A Case Study at the Ministry of Justice, Peru

Fernando Rodríguez-Sánchez[1], Javier Gamboa-Cruzado[1], Liliana Chanona-Hernández[2,*]

[1] Universidad Nacional Mayor de San Marcos,
Facultad de Ingeniería de Sistemas e Informática,
Peru

[2] Instituto Politécnico Nacional, ESIMEZ,
Mexico

{fernando.rodriguez13, jgamboac}@unmsm.edu.pe, lchanonah@ipn.mx

**Abstract**. The current management of radio-electronic signal blocking systems in penitentiary environments faces significant challenges, such as inefficiency in signal disruption, non-compliance with authorized signal standards, and the generation of social tensions in nearby communities. The purpose of this study is to apply Best Practices that refine the operability of such systems in penal centers, in order to minimize unauthorized communications from these facilities, optimize signal penetration within them, and mitigate protests from the surrounding population. To this end, the FRS Methodology based on the PMBok methodology, as well as the PDCA Method and the ITIL Methodology were created and utilized; a case study was conducted in a group of penal establishments in Peru, where the methodology has been implemented. In this paper, three key indicators have been addressed, and through the implementation of the FRS methodology, the established objectives have been achieved. The results obtained exhibit behavior within the expected range, allowing us to conclude that the FRS methodology can improve the management process of radio-electronic signal blocking systems in penal institutions.

**Keywords**. Best practices, jamming, interference, radio-electronic signal blocking, penal establishments, downlink.

## 1 Introduction

The current management process of radio-electronic signal blocking systems faces several challenges. One of the most significant problems is inefficiency in signal blocking, resulting in failures of the blocking system and the occurrence of unauthorized calls from within penal institutions.

Moreover, compliance with the allowed signal levels inside the prisons as per the current regulations is not achieved, leading to frequent requests to the authorities to conduct radiofrequency signal measurements in the untouchable zone and to verify compliance with these standards.

Another critical aspect is the generation of social conflicts in communities near the prisons due to the negative impact of the blocking systems on public telecommunications services. Blocking techniques must impact the quality parameters of the signal. For example, in the case of LTE, according to the research conducted by Mahendra, Budiman, Cahyadi, and Taruk [1], blocking techniques can affect measures such as Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), and Signal to Noise Ratio (SNR).

Regarding the blocking techniques themselves, Samir and Shaheen [2] mention several models of blocking signals, including Barrage, Band Jamming (BJ), Partial Band Noise Jamming (PBJ), Pulsed Jamming Technique, and a combination of the last two called Partial-Pulsed Band jamming (PPBJ).

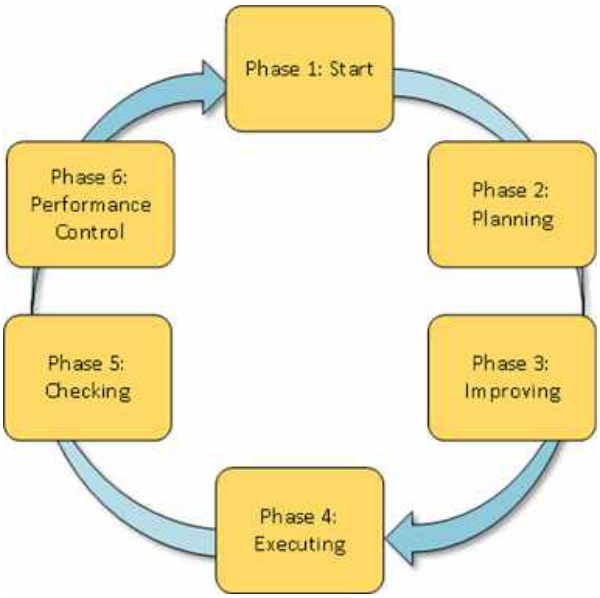



**Fig. 1.** FRS methodology diagram

Jasim and Kamil [3] also highlight different techniques of Jamming attacks, such as Direct Jamming, Flooding Jamming, Selective Jamming, and Denial of Service (DoS) Jamming. Concerning the goal of jamming attacks, Mert and Gunes [4] suggest that for LTE, targeting the main synchronization signals (primary and secondary, PSS and SSS) in the Downlink is an effective strategy.

On the other hand, El-Keji, Ureten, Yanikomeroglu, and Yensen [5] indicate that LTE is vulnerable to jamming and that the most susceptible control channel in the Downlink is the Physical Downlink Control Channel (PDCCH). These findings underscore the importance of understanding and mitigating jamming threats in LTE networks. Different approaches have been discussed regarding interference in LTE systems. Bazzo, Takaki, Barros, and Carrillo [6] suggest that mitigation techniques are available for both Downlink and Uplink in the LTE system.

However, Mohamed Elatif Soliman [7] raises that partial band interference may be more effective than the barrier interference technique. It is also highlighted that the most effective location for the interference signal is at the center of the six RB (resource blocks) where the synchronization channels (PSS and SSS) and PBCH are located.

On the other hand, Eduard Krutisky [8] emphasizes that launching an interference attack can be easier than mitigating it, although it is possible to develop countermeasures, but these do not guarantee complete protection of all networks.

Ali, Baddeley, Bariash, Lopez, Lunardi, and Giacalone [9] conducted an analysis of the error rate performance in WLAN IEEE 802.11n communication systems in the presence of different types of interferences.

The aforementioned studies present various approaches to the vulnerability of LTE to jamming attacks. Vo-Huu, Vo-Huu, and Noubir [10] highlight an interference strategy that can have a significant impact, destroying more than 95% of the transmitted packets with only 0.1% of the regular transmitted signal power.

Marc Lichtman, Roger Piqueras, Mi.na Labib, Vuk Marojevic, and Jeffrey Reed [11] point out that LTE is vulnerable to jamming attacks because it was not originally designed to be a critical communications technology. Conversely, Rafal Krenz and Soumya Brahma [12] warn about the vulnerability of LTE to be jamming attacks and denial of service (DOS) attacks, which can effectively disrupt communication.

Vuk Marojevic, Jeffrey Reed, and Raghunandan Rao [13] highlight the threat posed by jamming attacks on the PSS/SSS channels to the performance of LTE after a device connects to a cellular base station (EBC). In addition, Ibrahim Patel, Ashok Shigli, and Srioathi Raja [14] managed to successfully experiment with jamming to inhibit phone calls.

Hossein Pirayesh and Huacheng Zeng [15] investigated jamming attacks on Wi-Fi and mobile networks, analyzing their PHY and MAC layers, and explored possible countermeasures, also addressing the vulnerability of LTE to jamming attacks, evaluating their PHY and MAC layers, and considering security measures. The study conducted by Yulong Zou, Jia Zhu, Xianbin Wang, and Lajos Hanzo [16] highlights the vulnerability of radiofrequency (RF) transmissions to jamming attacks. This is due to the nature of radio propagation, which can be susceptible to external interference, in addition, security at the physical layer of wireless communications can be compromised by effective jamming attacks. According to Victor Gil Jimenez, Ana Garcia

**Table 1.** Operationalization of the Dependent Variable: Management of Radio-Electronic Signal Blocking Systems

| Indicator | Index | Unit of Measure | Observation Unit |
|---|---|---|---|
| Number of events or complaints of failures in the blocking of the EPs | [1-24] | Events | Reports, indirect observation |
| Number of requests to the MTC to perform measurements | [1-24] | Requests | Reports, indirect observation |
| Number of social protests in the vicinity of the EP | [1-24] | Protests | Reports, indirect observation |

Amada, and Francisco Hernando Gallego [17], orthogonal sequences have been developed to interfere with UMTS transmission signals, reducing the energy required to carry out a jamming attack.

Furthermore, Grecia Romero, Virginie Deniau, and Olivier Stienne [18] observe that the error rate in jamming attacks on Wi-Fi signals with protocol 802.11n can vary according to the waveform of the interference signal. Regarding the 5G NR (Fifth Generation) technology, Marc Lichtman, Vuk Marojevic, and Jeffrey Reed [19] note that it is less vulnerable to interference compared to LTE, due to its dynamic nature and the elimination of dispersed control channels.

However, Youness and Saleh [20] mention that 5G NR still uses primary and secondary synchronization signals (PSS and SSS), and Skokowski, Kelner, Malon, Maslanka, Birutis, Vasquez, Saha, Low, Czapiewska, Magiera, Rajchowski, and Ambroziak [21] assert that an effective jamming attack on 5G NR targets these synchronization signals, the physical broadcast channel (PBCH), and the downlink control channel (PDCCH) at the physical layer (PHY).

It is public knowledge that complaints persist about the making of unauthorized telephone calls from within different penal establishments. Therefore, it is essential to improve the efficiency indicators of the radio-electric signal blocking. Moreover, the Ministry of Transport and Communications (MTC) has conducted numerous radiofrequency (RF) signal measurements because both the service provider and mobile operators do not comply with the signal level limits established by legal regulations.

This highlights the need to establish more agile coordination lines for the regulation of these aspects. This has become even more pronounced during the pandemic era when classes and work were conducted virtually. The goal of implementing Best Practices is to optimize the Management of the Radio-Electric Signal Blocking System in Penal Establishments, using the FRS methodology.

The article is structured as follows: section 2 presents the Theoretical Background, where a synthesis of the relevant theory is offered. Section 3 addresses the Research Method used. The development of the new Collaborative Model SKX and its phases are presented in section 4. Section 5 is dedicated to the Case Study conducted in Peru. In section 6, the Results are presented and discussed in detail. Finally, section 7 presents the Conclusions and outlines areas for future research.

## 2 Theoretical Background

This section provides a solid and essential context for understanding the implementation of Best Practices in the Management of Radio-Electronic Signal Blocking Systems at the Ministry of Justice in Peru. It explores historical and conceptual backgrounds related to signal blocking technology.

### 2.1 Best Practices

In the context of managing radio-electronic signal blocking systems, it has been deemed best practice to block or jam signals on the Downlink. This includes synchronization signals PSS, SSS, the physical broadcast signal (PBCH), and the control channel (PDCCH), in that order of priority, according to various studies by Nieves and Wyglinski [22], Youness and Saleh [20], among others. In addition, it has been determined that the jamming signal should be centered on the carrier. Regarding best practices related to the prison environment, the publication GSMA Latin American [23] has identified the following recommendations: identifying the penitentiary establishments that truly require a radio-electronic


**Fig. 2.** Project Management Processes, PMBok

signal blocking system, analyzing and defining the technical solution for each prison, taking into account its individual characteristics and needs, coordinating with radio spectrum authorities and mobile operators to define signal levels, and carrying out controlled tests.

### 2.2 Management of Radio-Electronic Signal Blocking Systems

Pavel Tomasek [24] has conducted a study in this regard, verifying that these arrangements cannot block magnetic fields, such as the Earth's magnetic field, but can protect the interior from electromagnetic radiation coming from outside. Farhan Aziz, Jeff Shamma, and Gordon Stuber [25] proposed an algorithm to estimate the type of interference with high probability and rapid convergence. Hadeel Obaid [26] pointed out that pulsed jammers can degrade network performance and cause a denial of service. Hossein Noori and Saeed Sadeghi [27] explored anti-jamming measures in two different scenarios, one where the target user has a lot of power and can block other users, and another where the target user has little power and does not affect the transmissions of others.

Birutis and Mykkeltveit [28] suggest that the most effective method to block the Downlink in 5G is to use Barrage Jamming (BJ) with high power in EIRP (Equivalent Isotropically Radiated Power), compared to blocking the synchronization signals (PSS and SSS). Regarding the deployment of 5G technology, Agiwal, Kwon, Park, and Hin [29] mention that the 3GPP standard allows dual connectivity using both 4G LTE and 5G NR at the same time.

This means that if an operator uses an LTE network infrastructure composed of the EPC core and LTE base stations (eNB), it can implement dual connectivity for the initial commercialization of 5G networks.

In this context, Fehmi, Fakhouri, Bahnasse, and Talea [30] explain that 5G NSA (Non-Standalone) is deployed in areas where there is already an existing LTE core and LTE radio access network (EPC).

## 3 Research Method

This section details the innovative experimental methodology used to assess the influence of radio-electronic interferences on various communication technologies. It also describes the phases of the Scientific Method adopted for its implementation and empirical validation. Collaborative Meta-Model: Agile FRS Methodology. The FRS Methodology represents an innovative approach designed to improve the current process of managing blocking systems in penal institutions. It comprises six key phases with the goal of contributing new knowledge and criteria to reduce the impacts caused by these systems in the


**Fig. 3.** PDCA cycle, plan-do-check-act

surrounding areas of prisons (See Figure 1). The phases of the new agile FRS methodology are:

**Phase 1.** Initiation, involves the process associated with the signing of the Unnamed Service Provision Contract "Provision of Technological Security Services in Prisons" (CIPS), for the provision of public fixed telephone service inside thirty-three (33) Penal Establishments (EP), receiving the service of blocking and/or inhibition of radio-electronic signals in the EPs, for 25 years. The deliverable in this phase is the signing of the Contract and the responsibilities assumed.

**Phase 2.** Planning, during this phase, specialized technical studies are carried out to assess the technical infrastructure that will be installed in each of the penal establishments.

Additionally, it includes the acceptance process by specialized personnel, who verify the mentioned studies.

**Phase 3.** Improvement, studies designed to significantly mitigate or prevent the blocking system equipment from generating negative interferences in the surroundings of the prisons will be carried out.

**Phase 4.** Execution, proceeds with the installation of all the technological infrastructure in the penal establishments, as described in the specialized technical studies. This includes the implementation of recommendations arising from studies of non-affectation to the population and interference with mobile operators. Likewise, exhaustive tests of the entire system are carried out, and the necessary verifications are made to ensure its correct operation.

**Phase 5.** Verification, Specialized personnel carry out exhaustive evaluations to determine if the blocking system affects public telecommunications services in the untouchable zone and if it causes interference with mobile operators. In case the results of these evaluations are favorable, and no significant negative impacts are detected.

**Phase 6.** Monitoring and Control, Throughout the operation period, from the beginning to the end of the contract, the blocking systems will be in full operation and will be subject to constant monitoring. Specialized personnel will carry out the monitoring and control of these systems within the penal establishments, while also evaluating any possible impact on public telecommunications services outside the prisons.

### 3.1 Applied Research Method

#### 3.1.1 Operationalization of Variables

Table 1 shows the operationalization of the dependent variable.

#### 3.1.2 Research Design

The research is classified under the experimental design focusing on the category of "pure" experiments. It involves a comparison of the experimental group (Ge) composed of randomly selected (R) Management Processes of Radio-Electronic Signal Blocking Systems, from which values $O_1$ are obtained. After that, the stimulus (X), the Best Practices according to the FRS methodology, is provided, and post-test results without the stimulus to the control group (Gc), $O_2$, are expected:

| | | | |
|---|---|---|---|
| $R$ | $G_e$ | $X$ | $O_1$ |
| $R$ | $G_c$ | $-$ | $O_2$ |

#### 3.1.3 Universe and Sample

The study universe comprises all the Management Processes of Radio-Electronic Signal Blocking Systems operating within penal establishments in Peru, with an indeterminate size $N$. For the sample, it focuses on those Radio-Electronic Signal Blocking Systems specifically linked to the Ministry of Justice and its respective penal establishments that are under the CIPS contract for the

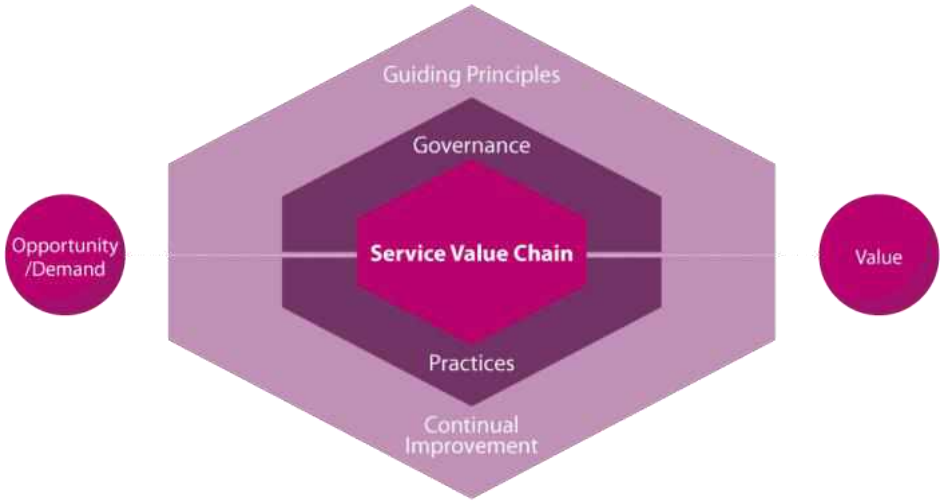



**Fig. 4.** Service value system structure

management of said systems, represented by a sample of $n = 30$. The sampling approach adopted is random, thus ensuring representativeness and randomness in the sample selection for the study.

### 3.1.4 Data Collection Procedure

Throughout the study, indirect observation methods were employed, and specialized databases were consulted. The collected data were analyzed using Minitab statistical software, with descriptive statistics applied to characterize each studied indicator. Additionally, for the scrutiny of the relationship between variables and the verification of hypotheses, inferential statistics techniques were used, employing the Student's t-test for the analysis of parametric nature indicators.

### 3.1.5 Hypothesis Statement

**H1:** If Best Practices are applied using the FRS methodology, it is expected that the number of events or complaints related to blocking failures will decrease, which would translate into an increase in the efficiency of the Management of the Radio-Electronic Signal Blocking System in Penal Establishments.

**H2:** If Best Practices are applied using the FRS methodology, it is expected that the number of requests to the MTC to perform signal measurements to detect signal levels above what is allowed will decrease in areas close to where the Radio-Electronic Signal Blocking System is installed in Penal Establishments.

**H3:** If Best Practices are applied using the FRS methodology, it is expected that the number of social protests in populations near the prisons, where the Radio-Electronic Signal Blocking Systems are installed in Penitentiary Establishments, will decrease.

For the contrast of the hypotheses, the following solution was proposed for each of the indicators:

$\mu_1$: Population mean (H1, H2, H3) for Post-Test of $G_c$

$\mu_2$: Population mean (H1, H2, H3) for Post-Test of $G_c$

where: Ho: $\mu_1 \leq \mu_2$, Ha: $\mu_1 > \mu_2$. Finally, the normality test of data, descriptive statistical analysis (See Figure 15, Figure 16, Figure 17, Table 3, and Table 4), and hypotheses were validated using the Student's t-test with the specialized software Minitab (See Table 5).

### 3.1.6 Development of the New Collaborative FRM Model

Next, the methods used to develop the new FRM methodology for the Management of radio-electronic signal blocking systems in penal facilities are presented.

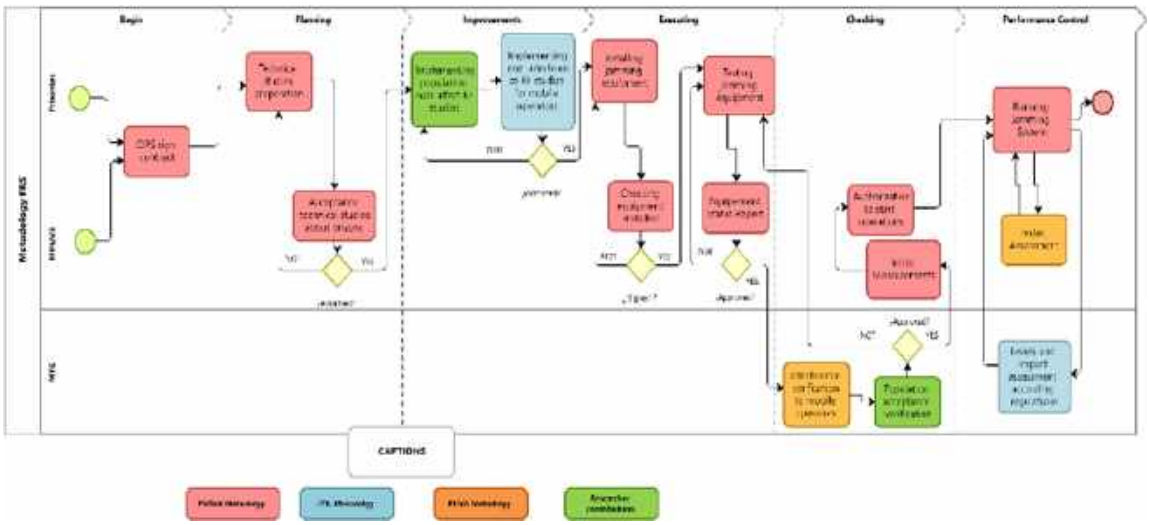


**Fig. 5.** FRS Methodology Flowchart

### 3.1.7 Reviewed Models

#### 3.1.7.1 Project Management Methodology based on PMBok

Project Management involves the strategic implementation of knowledge, skills, tools, and methodologies on project tasks to meet the stipulated requirements [31]. This process manages tasks with established schedules and concrete goals, as illustrated in Figure 2.

**Phase 1.** Initiation Processes: According to the authors [31], at this stage, the organization establishes the project's objectives, identifies the main stakeholders, appoints the Project Manager (PM), and grants formal authorization to initiate the project [32].

**Phase 2.** Planning Processes: [31] state that stakeholders play a crucial role in defining the project's scope and setting its objectives. At this stage, the work team develops a plan that will serve as a guide to achieve project success.The planning processes follow an integral approach, ensuring detailed planning before starting the project execution to produce the final product [32].

**Phase 3.** Execution Processes: [31] indicate that the project manager plays a key role in coordinating all the necessary resources to launch the project management plan.

This is where the project work is executed, using most of the allocated resources, time, and budget to achieve the project's objectives and the estimated business value [32].

**Phase 4.** Monitoring and Control Processes: [31] affirm that the project manager and their team are continuously monitoring the project's progress and taking corrective actions when necessary.

This stage focuses on evaluating real and quantitative data on how the project is developing compared to the original plan and determining the measures to address any variation that exceeds acceptable limits [32].

**Phase 5.** Closing Processes: [31] mention that the client formally accepts the project deliverables, which is essential since all projects must have a formal closure, regardless of whether they have been completed or not. The project closing processes involve administrative activities, such as collecting and finalizing all the necessary documentation to complete the project [32].

#### 3.1.7.2 PDCA Cycle (Plan, Do, Check, Act)

Also known as Deming's Cycle, it consists of four phases: Plan, Do, Check, Act, as shown in Figure 3.

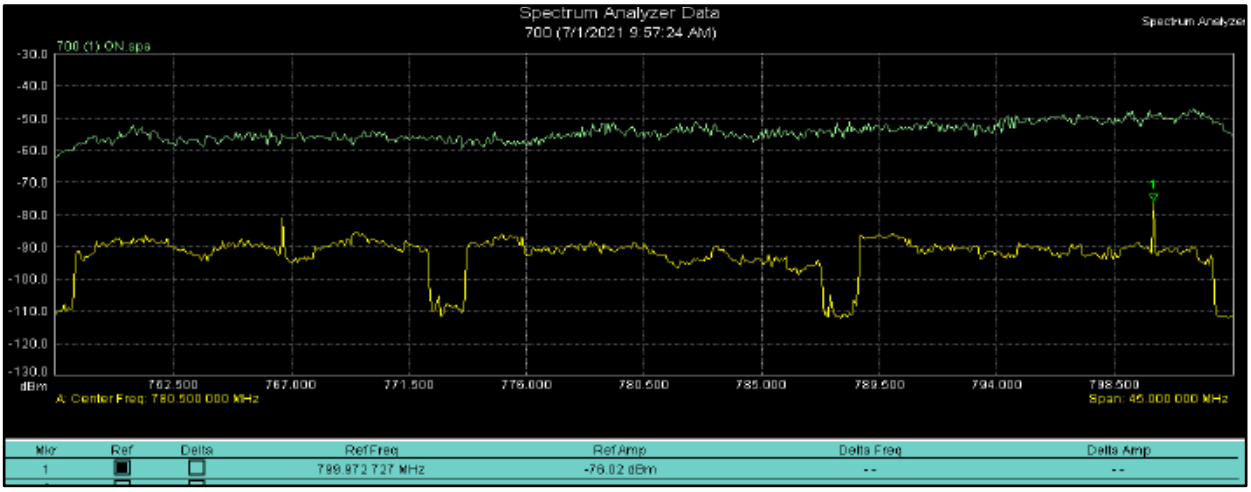



**Fig. 6.** Original LTE and blocking signal

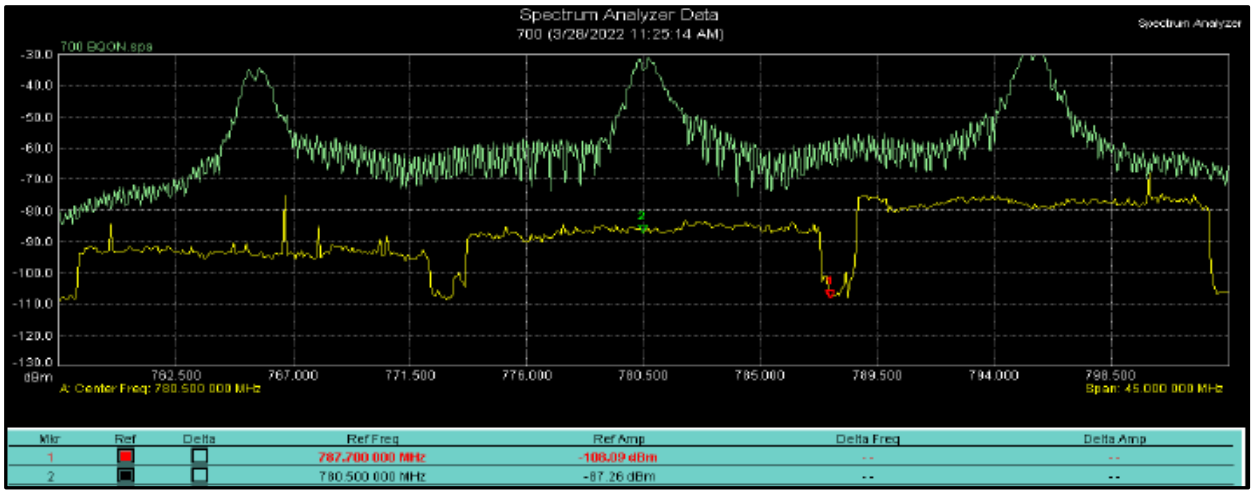



**Fig. 7.** Modified LTE and blocking signals

- **Phase 1.** Plan: Where the system's objectives and processes are established, as well as identifying and allocating the necessary resources to generate the desired outcomes [33].
- **Phase 2.** Do: Where tasks are executed and implemented as planned in Phase 1.
- **Phase 3.** Check: Essential for carrying out the monitoring and measurement of the processes and the resulting products and services compared to what was planned [33].
- **Phase 4.** Act: In this phase, it is crucial to take actions to improve performance and leverage opportunities for improvement. Actions necessary to correct unwanted effects and ensure that the project progresses as planned should also be implemented [33].

#### 3.1.7.3 ITIL Methodology

Provides a comprehensive digital operational model covering the entire organizational spectrum from start to finish, efficiently managing the

**Table 2.** Frequencies of blockers

| Band | Jamming Range (MHz) |
|---|---|
| 700 MHz | 758-803 |
| 850 MHz | 851–894 |
| 900 MHz | 939–960 |
| 1900 MHz | 1930–1990 |
| 2100 MHz | 2110–2150 |
| 2300 MHz | 2360–2390 |
| 2.4 GHz [Wi-Fi] | 2400–2483.5 |
| 2600 MHz | 2570–2615 TDD |
| | 2615–2690 FDD |
| 3500 MHz | 3500–3600 TDD |
| 5 GHz [Wi-Fi] | 5150–5850 |
| 1500 MHz | 1520–1570 / 1610-1675 |

provision and operation of IT-powered products and services [34], as detailed in Figure 4.

- **Phase 1,** Plan: Ensures that all involved parties have a common understanding of the vision, the current situation, and the direction towards improvement in the four dimensions [34].
- **Phase 2,** Improve: Promotes the constant improvement of products, services, and practices at all stages of the organization's value chain [34].
- **Phase 3,** Engage: Encourages a solid understanding of stakeholder needs, promotes transparency, and maintains ongoing commitment [34].
- **Phase 4,** Design and Transition: Ensures that products and services meet the stakeholders' quality expectations, costs, and market times [34].
- **Phase 5,** Obtain and Build: Ensures that service components are available at the required time and place [34].
- **Phase 6,** Deliver and Support: Ensures that services are delivered and supported according to the agreed specifications [34].


**Fig. 8.** Blocker and hardware to reduce propagation

#### 3.1.8 Collaborative Meta-Model: FRM Methodology

After thorough analysis of existing models, an innovative model has been developed, whose flowchart is presented in Figure 5.

## 4 Case Study

The FRS methodology presents a proposal aimed at improving the current method used in the installation of Blocking or Inhibition Systems in penal establishments.

### 4.1 Phase 1: Initiation

Involves the process associated with the signing of the Unnamed Service Provision Contract "Technological Security Services in Prisons" (CIPS), for the provision of fixed public telephone service inside 33 Penal Establishments (EP), receiving the service of blocking and/or inhibition of radio-electronic signals in the mentioned EPs, for 25 years. The deliverable in this phase isthe signing of the Contract and the responsibilities assumed.

### 4.2 Phase 2: Planning

The Provider is responsible for carrying out technical studies related to all the technological infrastructure that will be installed in the penal establishments. These studies cover fundamental aspects, such as the blocking system itself, including equipment, transmitters, antennas, and support structures.

They also encompass the power system, which includes electrical wiring, the generator set, and the electrical distribution, as well as the communications room and the conduit needed for electrical and telecommunications wiring. This document, called Specialized Technical Studies (ETE), is one of the main deliverables of this phase and is sent to MINJUS for review and, if applicable, approval.

### 4.3 Phase 3: Improvement

Studies specifically designed to significantly mitigate or completely avoid the blocking system equipment from generating negative interferences in the surroundings of the prisons will be carried out.

This includes both public telecommunications services and the prevention of interference in mobile operators' base stations, that is, without causing inconvenience or negatively affecting the surrounding telecommunications networks.

As a reference, one can resort to the theoretical study where an LTE signal (2.6 GHz) of 5 MHz RF width is blocked with a jamming signal of 465 KHz, according to the study [12], attached as a practical example is the blocking of an LTE signal 700 MHz, 15 MHz RF width, with a jamming signal of 2 MHz, according to Figure 6 and Figure 7.

Additionally, as shown in Figure 8, an example of the hardware required to be installed on the antennas of the blockers to minimize propagation to the nearby population is presented. Also, specific operating frequencies for blocking public telecommunications services in penal establishments are available, usually blocking on the Down Link, as indicated in Table 2.

### 4.4 Phase 4: Execution

Exhaustive research has been conducted to ensure that the implementation of the infrastructure does not have adverse impacts on the surrounding population or cause interference with mobile operators' services. Subsequently, the installation of such infrastructure in penal facilities progresses, adhering to the meticulously outlined guidelines in the specialized technical studies.

These studies include analysis on the non-affectation to the community and the absence of interference with mobile operators, culminating in the production of illustrated deliverables, as shown in Figure 9. Afterward, the installation of all the infrastructure mentioned in Phases 2 and 3 was verified. Once the conformity of the installation of all the technological infrastructure was confirmed, the Equipment Installation Act of the respective penal establishment will be signed, as well as the following tests:

- The blocking within the penal establishment.
- The non-affectation to the population within a distance of 200 meters from the perimeter of the penal establishments, defined as an untouchability zone, of the public telecommunications services.
- The non-interference with mobile operators, which involves measurements beyond 200 meters.

During this critical phase of the process, it's important to note that multiple instances of feedback from other stages or phases within the FRS methodology may emerge. This is due to the performance of essential calibrations aimed at achieving the pre-established objectives.

Consequently, various adjustments may be needed to ensure the achievement of the defined goals. MINJUS sector experts will be responsible for conducting technical evaluations following their established protocol to verify the effectiveness of signal blocking within the penitentiary facility:

- Testing the wireless public telecommunications services associated with the frequencies of the existing 2G, 3G, 4G, and 5G technologies.

**Equipment of the blocking system**

| Item | Cantidad | Equipo | Marca | Modelo | N Serie |
|---|---|---|---|---|---|
| 1 | 1 | Router | Tangreat | DTM-330A | - |
| 2 | 1 | Servidor Bloqueo | ThinkSystem | 7Y49AD41LA | J1019NB7 |
| 1 | 1 | Monitor Bloqueo | LG | 19M38A | 103NTRL9705-4 |
| 4 | 1 | Mouse | Logitech | Logitech Mouse K120 | HSD4BHB |
| 5 | 1 | Teclado | Logitech | Logitech Keyboard K120 | 20485C33N5K8 |

**Equipment of the blocking system - cases**

| Item | Equipo. | Marca | Modelo | Bandas | Frecuencia | N' Serie |
|---|---|---|---|---|---|---|
| 1 | Unidad de Bloqueo o inhibicion DQ-00 | Tangreat | TG-101M.NET | 5 | 1500<br>700<br>850<br>2100<br>2600 | 210530101M.NET263 |
| | | Tangreat | TG-SSC | 2 | 900<br>1900 | 210535303M0258 |
| | | Yonlit | Jammer | 1 | 5000 | HL20210622P01A03 |
| 2 | Unidad de Bloqueo o Inhibicion DQ-02 | Tangreat | TO-201M.NET | 4 | 700<br>850<br>2100<br>2600 | 210520101M.NET256 |
| | | Tangreat | TO-SSC | 1 | 1900 | 210535103M0259 |
| 3 | Unidad de Bloqueo o Inhibicion BQ-01 | Tangreat | 10-101M.NET | 4 | 700<br>850<br>2000<br>2000 | 210520101M.NET267 |
| | | Tangreat | TG-SSC | 2 | 900<br>1900 | 210515101ML0260 |
| 4 | Unidad de Bloqueo o Inhibicion DQ-04 | Tangreat | TO-101M.NET | 5 | 700<br>850<br>2100<br>2400<br>2000 | 210530101M.NET268 |
| | | Tangreat | TG-SSC | 1 | 1900 | 210515101ML0261 |
| 5 | Unidad de Bloqueo o Inhibicion DQ-05 | Tangreat | TG-101M.NET | 4 | 700<br>850<br>2100<br>2000 | 210530101M.NIT20S |
| | | Tangreat | TG-SSC | 2 | 900<br>1900 | 210520101M.NET269 |
| 6 | Unidad de Bloqueo o Inhibicion BQ-06 | Tangreat | 10-101M.NET | 5 | 700<br>850<br>2100<br>2400<br>2600 | 210520131M.NRT220 |
| | | Tangreat | TG-SSC | 1 | 1900 | |
| 7 | Unidad de Bloqueo o Inhibicion BQ-07 | Tangreat | 10-101M.NET | 6 | 300<br>$50<br>1900<br>2100<br>2100<br>2000 | 210422101M.NIT247 |
| | | Tangreat | TG-SSC | 2 | 900<br>1900 | 210515101ML0264 |

**Fig. 9.** Equipment list

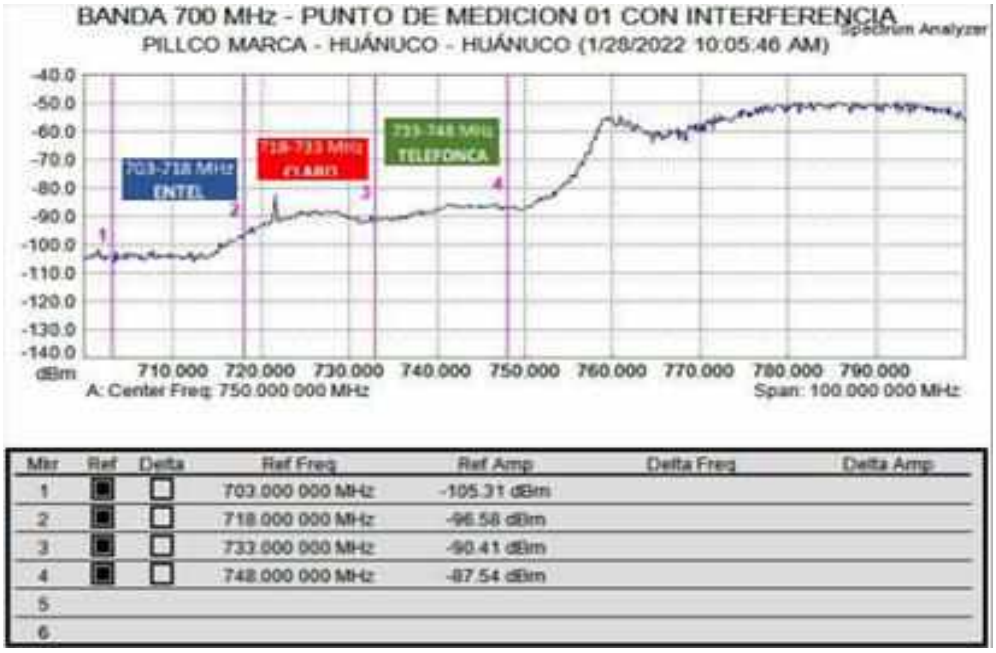



**Fig. 10.** MTC reports, measurement point 1 with Interference

- Testing the satellite telephony services of the main existing services: Inmarsat and Iridium.
- Testing Wi-Fi signals in the 2.4 GHz and 5 GHz bands.

### 4.5 Phase 5: Verification

Specialized personnel from the Ministry of Transport and Communications (MTC) will conduct verifications to determine if there are impacts on public telecommunications services in the untouchability zone due to the blocking system, as well as interference with mobile operators. These verifications are carried out in accordance with R.M. 954-2016-MTC/01.03. If the results of the verifications are favorable, that is, no impacts are detected, the corresponding operation permits will be granted. In this phase, the following tests are performed: Verify if the blocking systems emit signals outside the range of Downlink frequencies, including all the blocking frequency bands.

Special attention is paid to the possibility that these signals are emitted in the Uplink range, which could indicate that the noise generated beyond the Downlink range has a reach that exceeds the untouchability zone, which is usually located more than 200 meters away.

The purpose of this verification is to assess the possibility of negative interference occurring at the base stations (BTS) of mobile operators, which can result in deliverables shown in Figure 10. MTC's specialized personnel determine if there is a significant impact on public telecommunications services in the areas surrounding the penal establishments, up to 200 meters from the untouchability zone. The main goal is to ensure that the population is served by any mobile operator or internet service provider.

The tests are performed following the Technical Protocol established in Ministerial Resolution 954-2016-MTC/01.03 by the MTC, as seen in Figure 11. The MTC, within the framework of its tests, generates technical reports that include the following reports, as seen in Figure 12.

After conducting the tests, and if it cannot be confirmed that there are no impacts on public telecommunications services in the untouchability zone around the prisons, and there are no interferences with mobile operators, the specialized personnel from MINJUS will proceed to

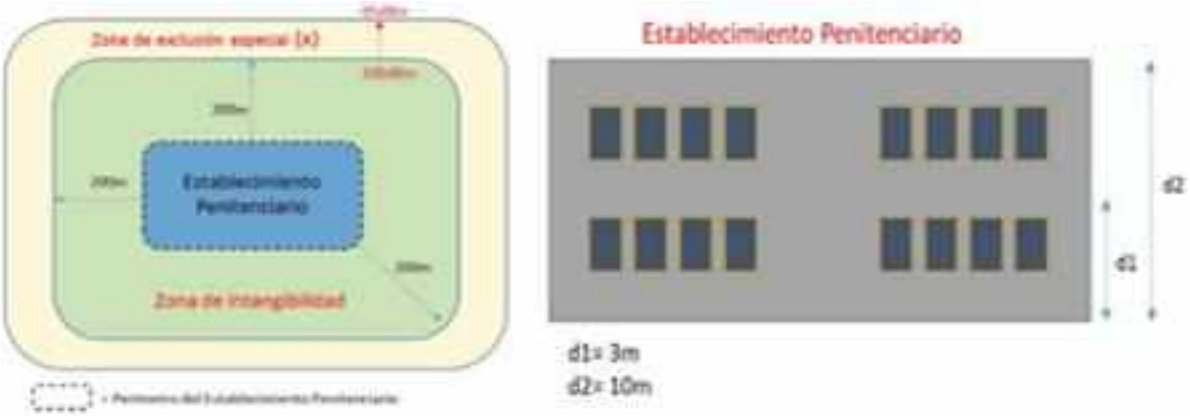



**Fig. 11.** Test zones

perform initial measurements of the radiofrequency (RF) signals.

This will allow for obtaining an "initial snapshot" of the situation of the RF levels of mobile operators operating within the prisons. Additionally, the MINJUS sector will create the corresponding reports to approve the start of operations of the blocking systems in the respective prison. The authorization to start operations is essentially a letter to the Provider, Prisontec S.A.C.

The initial snapshot, generated by the specialized MINJUS personnel, is the deliverable shown in Figure 13.

### 4.6 Phase 6: Monitoring and Control

After completing all the test phases, performing the relevant verifications, and obtaining the necessary authorizations, this stage, which spans from the start of operations to the end of the contract, will mark the beginning of the full operability of the blocking systems. These systems will be subject to constant monitoring, carried out by both specialized personnel from MINJUS and MTC.

This meticulous monitoring will cover the management and control of the blocking system within the penal facilities, as well as the analysis of possible impacts on public telecommunications services and interferences to the base stations of mobile operators located in the vicinity of the penal establishments.

## 5 Results and Discussion

This section presents the results obtained from the implementation of the FRS methodology in penal establishments and discusses these results in detail. The effects of applying Best Practices on radio-electronic signal blocking systems and their impact on the systems' efficiency, as well as on mitigating negative impacts on the population and mobile operators, are examined.

Additionally, RF signal measurements are analyzed, and it is evaluated whether the blocking systems comply with regulations to avoid generating interferences at mobile operators' base stations.

### 5.1 Results

Following the implementation of the proposed solution, specific values have been recorded for the following indicators: Number of Events of failures in the blocking of the EPs, Number of requests to the MTC for measurements, and Number of social protests in the vicinity of the EP.

### 5.2 Experimental Results: Reduction of I1, I2, and I3

30 values for each indicator were collected using various statistical techniques, the results of which are detailed in Figure 14. This data collection provides a broad and detailed view of the

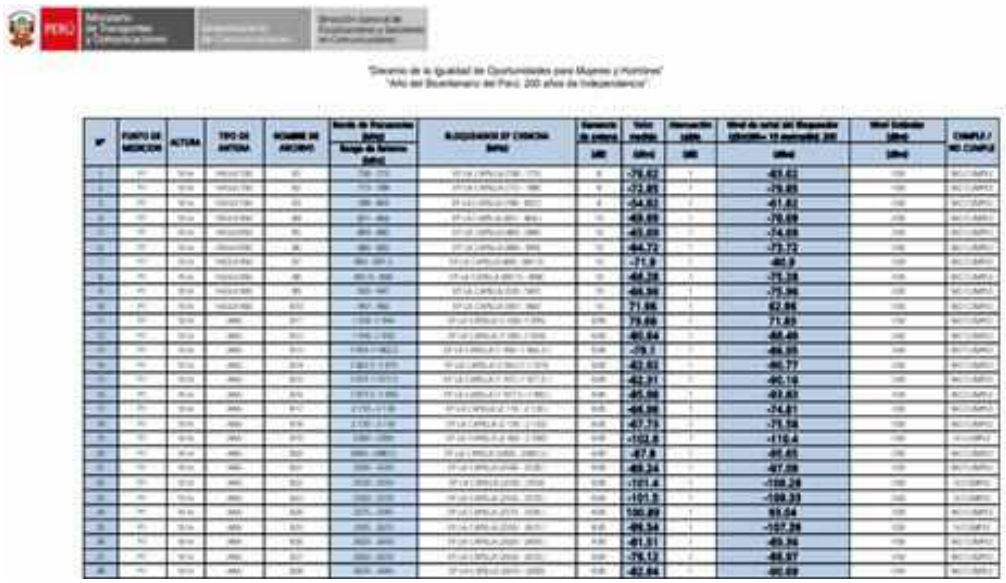


**Fig. 12.** Example of the MTC Report

performance and implications of the implemented measures, allowing for a rigorous evaluation of their effectiveness in the studied context.

### 5.3 Normality Test

This test focuses on comparing the Empirical Cumulative Distribution Function (ECDF) derived from the sample data and the theoretical distribution expected under the assumption of normality.

The evaluation is performed for three key indicators, providing a quantitative measure of how well the observed data fit a normal distribution model. (See Figure 15, Figure 16, and Figure 17).

I1: Number of Event Failures in the Blocking at Penal Establishments (EPs)

For this indicator I1, in the tests for the control group (Gc) and the experimental group (Ge), the results are: 0.237 and 0.090, respectively, both are $> \alpha$ (0.05). Therefore, the indicator values display a normal behavior.

I2: Number of Requests to the MTC for Measurements

For this indicator I2, in the tests for the control group (Gc) and the experimental group (Ge), the results are: 0.475 and 0.117, respectively, both are $> \alpha$ (0.05). Therefore, the indicator values display a normal behavior.

I3: Number of Social Protests in the Vicinity of the EPs

For this indicator, in the tests for the control group (Gc) and the experimental group (Ge), the results are: 0.161 and 0.052, respectively, both are $> \alpha$ (0.05). Therefore, the indicator values display a normal behavior.

### 5.4 Discussion of the Results

The discussion is carried out with a critical and honest approach by the researchers, who commit to respecting the integrity of the original results. This involves an ethical commitment not to alter or distort the information obtained during the research.

#### 5.4.1 With Descriptive Statistics

A preliminary analysis of the data is conducted, facilitating the identification of defined patterns within the collected information set. The specific details of this analysis are presented in Table 3 and Table 4. According to the results obtained from the

| Band | Operator | Operation Band (MHz) | Point 1 dBm | Band | Operator | Operation Band (MHz) | Punto 1 dBm |
|---|---|---|---|---|---|---|---|
| 700 MHz | Entel Perú S.A. | 758 – 773 | -116.62 | 1900 MHz | Entel Perú S.A. | 1962.5 – 1966.1 (2G) | -93.72 |
| | América Móvil Perú S.A.C. | 773 – 788 | -104.61 | | | 1966.1-1970.2(3G) | -110.76 |
| | Telefónica del Perú S.A.A. | 788 – 803 | -95.83 | | | 1970.2 – 1975 (3G) | -109.46 |
| IDEN | Entel Perú S.A. | 851 – 866 | -117.99 | | América Móvil Pe | 1975 – 1977.5 (2G) | -109.65 |
| 850MHz | Telefónica del Perú S.A.A. | 869 – 869.6 (2Gx3) | -98.5 | | Viettel Perú S.A.C | 1977.5-1981.5(3G) | -109.59 |
| | | 869.6 – 874.6 (3G) | -96.23 | | | 1981.5-1985.6(3G) | -108.68 |
| | | 874.6 – 874.8 (2Gx1) | -94.3 | | | 1985.6-1990(3G) | -109.74 |
| | | 874.8 – 879.8 | -103.85 | 2100 MHz | Telefónica del Pe | 2110 – 2130 | -109.04 |
| | | 890 – 891.5 (2G) | -91.3 | | Entel Perú S.A. | 2130 – 2150 | -109.79 |
| | América Móvil Perú S.A.C. | 880 – 885 (3G) | -106.46 | 2600 MHz | América Móvil Pe | 2505 – 2692 | -108.74 |
| | | 885 – 890 (3G) | -105.97 | 2.4 GHz | Wi-Fi | 2400 – 2484 | -114.61 |
| | | 891.5 – 894 (2G) | -94 | 3.5 GHz | Entel Perú S.A. | 3500 – 3525 | -103.2 |
| 900 MHz | Viettel Perú S.A.C. | 944-950 | -94.04 | | Telefónica del Pe | 3525 – 3550 | -106.7 |
| | | 950 – 960 | -97.68 | | Americatel Peru | 3550 – 3575 | -105.3 |
| 1900 MHz | América Móvil Perú S.A.C. | 1930 – 1935 (3G) | -110.66 | | Américal Móvil | 3575 - 3600 | -101.8 |
| | | 1935 – 1945 (LTE) | -107.75 | 5 GHz | Wi-Fi | 5150 – 5850 | -113.7 |
| | Entel Perú S.A. | 1945 – 1950 (3G) | -114.58 | 1500 MHz | Satelital | 1500 – 1700 | -105.55 |
| | Telefónica del Perú S.A.A. | 1950 – 1951.5 (2GX3) | -102.56 | | | | |
| | | 1951.5-1955.6(3G) | -108.25 | | | | |
| | | 1956.5-1960.6(3G) | -106.05 | | | | |
| | | 1960.6 – 1962.5 (2G) | -98.11 | | | | |

**Fig. 13:** MINJUS deliverable

Anderson-Darling normality test, both the AD value and the p-value turned out to be greater than the significance level α (0.05); therefore, the normality of the analyzed data is confirmed.

This indicates that, with a 95% confidence level, both the mean and the standard deviation of the data corresponding to the indicators reflect a normal distribution. This finding is fundamental for the use of parametric statistical techniques in subsequent analyses.

For each indicator presented in the table, it is observed that approximately 95% of the values are within a range of up to two standard deviations from the mean, suggesting a consistent and predictable data distribution. The kurtosis, indicating low peaks, suggests that the distribution is relatively flat, implying fewer extreme values.

On the other hand, the skewness towards lower values indicates that most of the data cluster on the lower side of the scale, and the third quartile (Q3) reveals that 75% of the values are less than or equal to this specific point.

**For Indicator 1**, the quantity of complaints about failures is how the efficiency of the blocking systems is evaluated. Regarding the efficiency of jamming equipment with 4G LTE radio-electronic signals of mobile operators, the work by Mert Eygi and Gunes Karabulut Kurt [4] indicates that if the synchronization signals for LTE follow the specifications of the 3GPP standard, in the subframe from 0 to 5, high levels of blocking efficiency are achieved.

The study by Mohamed Samir Elatif Soliman [7] has shown that partial band interference is more effective at the center where the synchronization

ISSN 2007-9737



| Nº | EP | I1. | | I2 | | I3. | | Nº | EP | I1. | | I2 | | I3. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Gc | Ge | Gc | Ge | Gc | Ge | | | Gc | Ge | Gc | Ge | Gc | Ge |
| 1 | EP de Chachapoyas | 1 | 0 | 1 | 0 | 1 | 0 | 16 | EP Mujeres de Chorrillos | 9 | 7 | 12 | 8 | 6 | 4 |
| 2 | EP de Chimbote | 5 | 3 | 3 | 2 | 4 | 2 | 17 | EP de Ancón (Ancón I) | 7 | 6 | 8 | 4 | 2 | 1 |
| 3 | EP de Huaraz | 4 | 4 | 4 | 3 | 3 | 1 | 18 | EP Modelo Ancón II | 6 | 5 | 7 | 1 | 3 | 2 |
| 4 | EP de Ayacucho | 9 | 7 | 10 | 7 | 12 | 10 | 19 | EP del Callao | 4 | 2 | 3 | 2 | 4 | 3 |
| 5 | EP de Cajamarca | 7 | 6 | 9 | 5 | 11 | 9 | 20 | EP de Huacho | 3 | 2 | 1 | 1 | 3 | 1 |
| 6 | EP de Cusco | 5 | 4 | 6 | 5 | 9 | 7 | 21 | EP de Huaral | 1 | 1 | 1 | 1 | 2 | 1 |
| 7 | EP de Huánuco | 4 | 2 | 5 | 4 | 6 | 4 | 22 | EP de Cañete | 1 | 0 | 1 | 0 | 1 | 0 |
| 8 | EP de Ica | 3 | 2 | 3 | 2 | 4 | 4 | 23 | EP de Puerto Maldonado | 3 | 3 | 8 | 7 | 10 | 8 |
| 9 | EP de Huancayo | 6 | 3 | 6 | 4 | 8 | 3 | 24 | EP de Juliaca | 7 | 6 | 9 | 6 | 9 | 7 |
| 10 | EP de Chanchamayo | 8 | 7 | 7 | 6 | 10 | 8 | 25 | EP de Puno | 2 | 1 | 6 | 3 | 5 | 4 |
| 11 | EP de Trujillo | 2 | 2 | 6 | 3 | 5 | 2 | 26 | EP de Tarapoto (Sananguillo) | 5 | 3 | 5 | 2 | 7 | 5 |
| 12 | EP de Chiclayo | 2 | 2 | 5 | 3 | 5 | 3 | 27 | EP de Moyobamba | 1 | 0 | 3 | 0 | 3 | 2 |
| 13 | EP de Lurigancho | 8 | 6 | 9 | 6 | 8 | 6 | 28 | EP de Tacna | 6 | 4 | 7 | 2 | 5 | 2 |
| 14 | EP Miguel Castro Castro | 10 | 8 | 11 | 8 | 7 | 6 | 29 | EP de Tumbes | 5 | 4 | 6 | 3 | 4 | 3 |
| 15 | EP de Chincha | 1 | 1 | 2 | 1 | 2 | 0 | | | | | | | | |

**Fig. 14.** MINJUS deliverable

channels and PBCH are located, managing to reduce the Bit Error Rate (BER) of 4G LTE signals to values of 0.5 and 0.22, when the standard value is 10 E-06.

Furthermore, according to the studies by Ali Abubakar et al. [9], the results of the most effective interference waveform and simulation results show a Packet Error Rate (PER) of 100% when employing Gaussian noise waveforms and modulated QPSK.

In addition, in the study by Triet Vo-Huu et al. [10], a new interference strategy with the IEEE 802.11 interleaving mechanism is developed to actively introduce burst errors in the Wi-Fi receiver, which can destroy more than 95% of the transmitted packets by using an interference power equivalent to only 0.1% of the normal transmitted signal power.

When the interference power increases to a fraction of 1%, this strategy can completely block all packets. Also, Samir and Shaheen [2] mention various blocking signal models, but it was shown that the proposed waveform interference technique completely deteriorates the BER performance of the OFDM system, which increases blocking efficiency and reduces the possibility of events and complaints of failures.

The application of the radio-electronic signal blocking scheme transcends its primary use in penal establishments, leading to its implementation in various contexts that require strict communication control to preserve integrity and security.

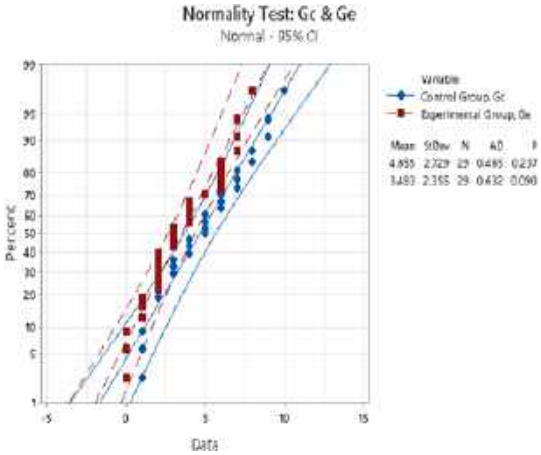



**Fig. 15.** Normality Test for Indicator I1

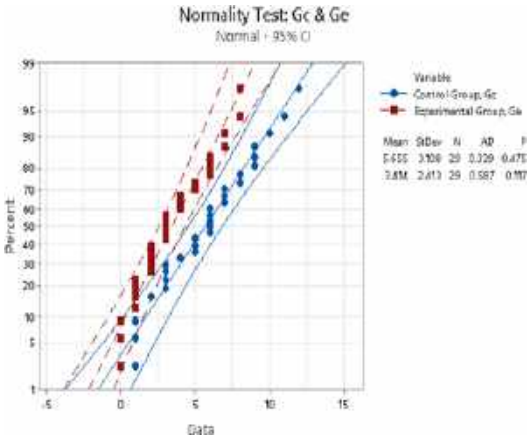



**Fig. 16.** Normality Test for Indicator I2

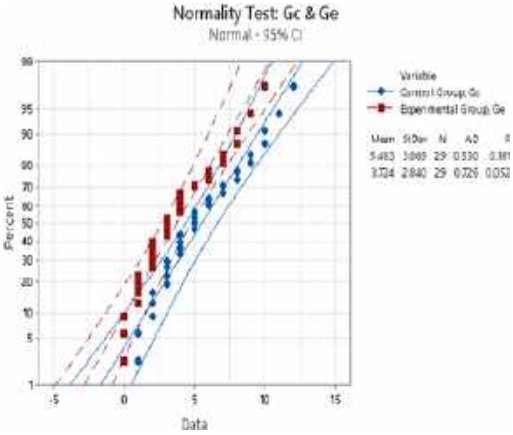



**Fig. 17.** Normality Test for Indicator I3

A notable implication of this approach is its potential application in university entrance exams, as well as in police and military training institutions, where it is critical to prevent fraud and ensure fairness. Similarly, the scheme emerges as a valuable tool for security in the banking sector, where signal blocking can prevent the misuse of electronic devices for illicit purposes.

**For Indicator 2**, the quantity of requests to the MTC to perform RF signal measurements is related to the intervention of the entity managing the radio-electric spectrum to regulate the activity of blockers as well as mobile operators around the prisons. Regarding RF signal measurement, the study by Oluyomi Simpson and Yuchuang Sun [35], who perform measurements of LTE signal received and measured from a base station (EBC) operating in the 2100 MHz band with a height of 15 m, a Tx power of 58.7 dBm and 2100 MHz, measured the following parameters:

Reference Signal Received Power (RSRP) in dBm, Reference Signal Received Quality (RSRQ) in dB, and Reference Signal Signal to Noise Ratio (RSSNR) in dB, with statistical values such as mean, standard error, median, mode, standard deviation, variance, kurtosis, etc.

According to the study by José Martínez Morales and Alvaro Casallas Hernández [36], in the Analysis of the Normative Impact of the National Spectrum Agency (ANE) and the Ministry of Information and Communication Technologies (MinTIC), who made adjustments to the blockers and despite the adjustments made by mobile operators, the National Penitentiary Institute of

Colombia (INPEC) stated that there is still a need to implement more robust technical measures that lead to attenuating as much as possible the emissions coming from the base stations within the prisons, to ensure the success of the radio-electric signal inhibition.

According to the study by Hsing-Yi and Tsung-Han Lin [37], using the Finite Difference Time Domain (FDTD) method is used to perform measurements simulating radiation patterns of an LTE base station antenna in vertical and horizontal planes at frequencies of 1795, 1920, and 2060 MHz respectively.

Also, according to the research conducted by Mahendra, Budiman, Cahyadi, and Taruk [1], the best deviation from Telkomsel operators is -64 dBm and the worst is -114 dBm.

The best point for XL operator is -69 dBm, the worst is -101 dBm, and for the best point of IM3 operator is -64 dBm, the worst is -104 dBm. According to Pramono, Alvionita, and Eko [38], the optimal levels for receiving LTE signal are: good (-80 to -90dBm), fair (-90 to -100 dBm), and poor (-100 to -110 dBm), expressed in RSRP (Reference Signal Reception Power) which is the amount of signal from the LTE station.

It is crucial to highlight the application of the FRS methodology as a significant implication both in the supervision of the current contract for the installation of blockers and in telecommunications projects in rural areas.

This methodology not only facilitates the effective implementation of signal blocking systems in penal environments but also promotes their use in the development of telecommunications infrastructure in less accessible areas, contributing to reducing the digital divide.

Additionally, the FRS methodology proves instrumental in determining optimal signal levels in the vicinity of prisons, ensuring that current legal regulations are effectively complied with.

**For Indicator 3**, the quantity of social protests or conflicts around the prisons, according to the study by Juan Pablo Parra and Carolina Botero [39], concludes that it is necessary for the State to create an independent mechanism that periodically reports the state of telecommunications to allow public scrutiny of internet quality and access.

**Table 3.** Results with Descriptive Statistics

| Sample | n | Mean | StDev | AD | p-value |
|---|---|---|---|---|---|
| I1: Post-Test (Gc) | 29 | 4.66 | 2.73 | 0.465 | 0.237 |
| I1: Post-Test (Ge) | | 3.48 | 2.35 | 0.632 | 0.09 |
| I2: Post-Test (Gc) | 29 | 5.66 | 3.11 | 0.339 | 0.475 |
| I2: Post-Test (Ge) | | 3.41 | 2.41 | 0.587 | 0.117 |
| I3: Post-Test (Gc) | 29 | 5.48 | 3.09 | 0.53 | 0.161 |
| I3: Post-Test (Ge) | | 3.72 | 2.84 | 0.726 | 0.052 |

**Table 4.** Summary of Results for the Indicators

| Sample | n | 95% CI for Mean | |
|---|---|---|---|
| I1: Post-Test (Ge) | 29 | 2.587 – 4.378 | |
| I2: Post-Test (Ge) | 29 | 2.496 – 4.331 | |
| I3: Post-Test (Ge) | 29 | 2.644 – 4.804 | |
| **Sample** | **Kurtosis** | **Skewness** | **Q3** |
| I1: Post-Test (Ge) | -1.0087 | 0.287 | 6 |
| I2: Post-Test (Ge) | -0.8357 | 0.4247 | 5.5 |
| I3: Post-Test (Ge) | -0.5435 | 0.642 | 6 |

**Table 5.** Hypothesis Testing for Parametric Indicators

| Sample | n | Ho | t-value | p-value |
|---|---|---|---|---|
| I1: Post-Test (Gc) | 29 | $\mu_1 \leq \mu_2$ | 1.75 | 0.043 |
| I1: Post-Test (Ge) | | | | |
| I2: Post-Test (Gc) | 29 | $\mu_1 \leq \mu_2$ | 3.07 | 0.002 |
| I2: Post-Test (Ge) | | | | |
| I3: Post-Test (Gc) | 29 | $\mu_1 \leq \mu_2$ | 2.26 | 0.014 |
| I3: Post-Test (Ge) | | | | |

They also recommend the State commit to not using signal jammers as much as possible with a policy respectful of rights, as well as requesting mobile operators (internet) in exercising their responsibility to users to analyze every request for blocking or interference.

The entity managing the radio-electric spectrum must investigate complaints about signal interference and internet access to protect citizens' rights.

In this line, the study by José Martínez Morales and Alvaro Casallas Hernández [36], in the Analysis of the Normative Impact of the National Spectrum Agency (ANE) and the Ministry of Information and Communication Technologies (MinTIC), pointed out complaints from citizens about failures in mobile communication services, and for this reason, some citizens have activated judicial mechanisms to seek to protect the fundamental right to communication, having as a

sentence in some cases the order to INPEC, the shutdown of the signal inhibition systems.

Also, in the study by Juan Cueva Silva and Giovane Mendieta Izquierdo [40], the social protest in Latin America in 2019 is not the same as in the early years of the new millennium or before, in the midst of this information and communication revolution, the interference of social networks in the dissemination of information is evidenced, with the use of the various services provided by the Internet, with the publication of videos and chats, as well as the manifestation of positions through Twitter, Facebook, and Instagram, without forgetting the chains in WhatsApp, has become the protagonist of a social protest that gets out of the hands of failed democratic systems and traditional media.

According to Pozo León [41], the installation of blockers in Ayacucho has experienced a loss in mobile communications and Wi-Fi in the population surrounding the prison, even its impact was greater during the COVID-19 pandemic, affecting virtual classes and remote work, which constitutes a negative impact, this research evidences the origin of the social protests that occurred in the place, as well as other places in the country.

In contrast, according to Villamizar and Sorzano [42], there was a belief that mobile phone antennas could produce lethal damage and that it was related to the spread of covid-19. Speculation and rumors about this relationship led to social mobilization and the demand for the dismantling of the antenna.

Finally, the application can be extended to the management of social conflicts derived from infrastructure and the provision of public goods and services.

Through the implementation of the FRS methodology, it is possible to effectively monitor the possible effects on the population, including not only public telecommunications services but also other essential public services that may be impacted.

This approach allows addressing the community's concerns and needs proactively, ensuring that the implemented measures not only meet technical and security objectives but also minimize the negative impact on citizens' quality of life.

### 5.4.2 With Inferential Statistics

In Table 5, the results derived from the application of statistical tests are presented, which have been used to effectively contrast the proposed hypotheses. These values are crucial for determining the validity of the hypotheses and, therefore, to support the conclusions of the research. Since all p-value results are less than α (0.05), the results provide evidence to reject the null hypothesis (Ho), and the alternative hypotheses were true, making the tests significant.

## 6 Conclusions and Future Research

The findings of this study lead to several conclusions. Firstly, it has been demonstrated that the use of Best Practices applying the FRS methodology decreases the number of events or complaints about failures in blocking, meaning it will increase the efficiency of the Radio-Electronic Signal Blocking System in Penal Establishments.

Another result of this study confirms that the use of Best Practices applying the FRS methodology reduces the number of requests to the MTC for signal measurements to detect signal levels above what is permitted, in areas near where the Radio-Electronic Signal Blocking System is installed in Penal Establishments.

Finally, one of the outcomes of this study is the reduction in the number of social protests in populations near the prisons, which occur due to the negative impact on public Telecommunications services, where the Radio-Electronic Signal Blocking Systems are managed.

A limitation of this study is not having reduced the events of failures to a greater extent. For future work, the study should include blocking of 5G NSA technology at an experimental level, as it has not yet been fully deployed for 5G NSA technology.

Also, the formation of an online monitoring system of the signals emitted by both mobile operators and blockers is suggested.

## References

1. **Putra, G. M., Budiman, E., Malewa, Y., Cahyadi, D., Taruk, M., Hairah, U. (2021).** 4G

LTE experience: reference signal received power, noise ratio and quality. 3rd East Indonesia Conference on Computer and Information Technology, pp. 139–144. DOI: 10.1109/eiconcit50028.2021.9431853.

2. **Wanis, M., Shaheen, E., Samir, M. (2018).** The impact of different jamming techniques on ofdm communication systems. The International Conference on Electrical Engineering, Vol. 11, No. 11, pp. 1–14. DOI: 10.21608/iceeng.2018.30147.
3. **Jasim, S., Kamil, O., Alhyani, N. (2018).** The impact of different jamming techniques on ofdm communication systems. International Journal Of Latest Technology In Engineering AND Management, Vol. 8, No. 1, pp. 16–22
4. **Mert, E., Kurt, K. (2020).** A countermeasure against smart jamming attacks on lte synchronization signals. Journal of Communications , Vol. 15, No. 8, pp. 626–632
5. **El-Keyi, A., Ureten, O., Yanikomeroglu, H., Yensen, T. (2017).** Lte for public safety networks: synchronization in the presence of jamming. IEEE Access, Vol. 5, pp. 20800–20813. DOI: 10.1109/access.2017.2751964.
6. **Bazzo, J. J., Barros, S., Takaki, R., Carrillo, D., Seki, J. (2017).** 4G/LTE networks for mission-critical operations: a cognitive radio approach. Telecommunications and Information Technology. Springer, Cham, Vol. 1, No. 5 pp. 51–64. DOI: 10.1007/978-3-319-53753-5_5.
7. **Shabana, G. H., Shaheen, E. M., Samir, M. (2021).** Emulation of optimized noise jamming impact on the lte downlink using systemvue. 10th International Conference on Intelligent Computing and Information Systems, pp. 105–115. DOI: 10.1109/icicis52592.2021.9694215.
8. **Krutitsky, E. (2022).** Cybersecurity - Introduction to Jamming. ResearchGate. DOI: 10.13140/RG.2.2.32263.47522.
9. **Ali, A. S., Baddeley, M., Bariah, L., Lopez, M. A., Lunardi, W. T., Giacalone, J., Muhaidat, S. (2022).** Jamrf: performance analysis, evaluation, and implementation of rf jamming over wi-fi. IEEE Access, Vol. 10, pp. 133370–133384. DOI: 10.1109/access.2022.3230895.
10. **Vo-Huu, T. D., Vo-Huu, T. D., Noubir, G. (2016).** Interleaving jamming in Wi-Fi networks. Proceedings of the 9th ACM Conference on Security and Privacy in Wireless and Mobile Networks, pp. 31–42. DOI: 10.1145/2939918.2939935.
11. **Lichtman, M., Jover, R. P., Labib, M., Rao, R., Marojevic, V., Reed, J. H. (2016).** LTE/LTE-a jamming, spoofing, and sniffing: threat assessment and mitigation. IEEE Communications Magazine, Vol. 54, No. 4, pp. 54–61. DOI: 10.1109/mcom.2016.7452266.
12. **Krenz, R., Brahma, S. (2015).** Jamming LTE signals. Proceedigns of the IEEE International Black Sea Conference on Communications and Networking, pp. 72–76. DOI: 10.1109/blackseacom.2015.7185089.
13. **Marojevic, V., Rao, R. M., Ha, S., Reed, J. H. (2017).** Performance analysis of a mission-critical portable lte system in targeted rf interference. IEEE 86th Vehicular Technology Conference, pp. 1–6. DOI: 10.1109/vtcfall.2017.8288187.
14. **Patel, I., Shigli, A., Raja, S., Kulkarni, R. (2018).** Intelligent 2G, 3G and 4G mobile signal Jammers. Proceedings of International Conference on Energy Efficient Technologies for Sustainability, pp. 1–8. DOI: 10.1109/vtcfall.2017.8288187.
15. **Pirayesh, H., Zeng, H. (2022).** Jamming attacks and anti-jamming strategies in wireless networks: a comprehensive survey. IEEE Communications Surveys and Tutorials, Vol. 24, No. 2, pp. 767–809. DOI: 10.1109/comst.2022.3159185.
16. **Zou, Y., Zhu, J., Wang, X., Hanzo, L. (2016).** A survey on wireless security: technical challenges, recent advances, and future trends. Proceedings of the IEEE, Vol. 104, No. 9, pp. 1727–1765. DOI: 10.1109/jproc.2016.2558521.
17. **Gil-Jiménez, V. P., Garcia-Armada, A., Hernando-Gallego, F., González-Pizarro, N. S. (2016).** Novel orthogonal multi-sequences for an efficient jamming on the umts signal. EURASIP Journal on Wireless Communications and Networking, Vol. 2016, No. 1, pp. 1–8. DOI: 10.1186/s13638-016-0770-6.

18. **Romero, G., Deniau, V., Stienne, O. (2019).** LTE physical layer vulnerability test to different types of jamming signals. International Symposium on Electromagnetic Compatibility, pp. 1138–1143. DOI: 10.1109/emceurope.2019.8872052.
19. **Lichtman, M., Rao, R., Marojevic, V., Reed, J., Jover, R. P. (2018).** 5G NR jamming, spoofing, and sniffing: threat assessment and mitigation. IEEE International Conference on Communications Workshops, pp. 1–6. DOI: 10.1109/iccw.2018.8403769.
20. **Arjoune, Y., Faruque, S. (2020).** Smart jamming attacks in 5g new radio: a review. 10th Annual Computing and Communication Workshop and Conference, pp. 1010–1015. DOI: 10.1109/ccwc47524.2020.9031175.
21. **Skokowski, P., Kelner, J. M., Malon, K., Maślanka, K., Birutis, A., Vazquez, M. A., Saha, S., Low, W., Czapiewska, A., Magiera, J., Rajchowski, P., Ambroziak, S. (2022).** Jamming and jamming mitigation for selected 5G military scenarios. Procedia Computer Science, Vol. 205, pp. 258–267. DOI: 10.1016/j.procs.2022.09.027.
22. **Flores, M. E., Poisson, D. D., Stevens, C. J., Nieves, A. V., Wyglinski, A. M. (2023).** Implementation and evaluation of a smart uplink jamming attack in a public 5g network. IEEE Access, Vol. 11, pp. 75993–76007. DOI: 10.1109/access.2023.3296701.
23. **GSMA. (2017).** Inhibidores de señal, uso de Jammers en prisiones. www.gsma.com/latinamerica/wp-content/uploads/2017/12/Reporte-Jammers-2017-Espan%CC%83ol.pdf
24. **Tomasek, P. (2016).** Attenuation of wireless communication under ieee 802.11ah. Annals of DAAAM and Proceedings, DAAAM International Vienna, pp. 838–844. DOI: 10.2507/27th.daaam.proceedings.121.
25. **Aziz, F. M., Shamma, J. S., Stuber, G. L. (2017).** Jammer-type estimation in lte with a smart jammer repeated game. IEEE Transactions on Vehicular Technology, Vol. 66, No. 8, pp. 7422–7431. DOI: 10.1109/tvt.2017.2672682.
26. **Obaid, H. S. (2020).** Wireless network behaviour during jamming attacks: simulation using opnet. Journal of Physics: Conference Series, Vol. 1530, No. 1, pp. 012009. DOI: 10.1088/1742-6596/1530/1/012009.
27. **Noori, H., Sadeghi-Vilni, S. (2020).** Jamming and anti-jamming in interference channels: a stochastic game approach. IET Communications, Vol. 14, No. 4, pp. 682–692. DOI: 10.1049/iet-com.2019.0637.
28. **Birutis, M. A., Mykkeltveit, A. (2022).** Practical jamming of a commercial 5g radio system at 3.6 GHz. Procedia Computer Science, Vol. 205, pp. 58–67. DOI: 10.1016/j.procs.2022.09.007.
29. **Agiwal, M., Kwon, H., Park, S., Jin, H. (2021).** A survey on 4G-5G dual connectivity: road to 5G implementation. IEEE Access, Vol. 9, pp. 16193–16210. DOI: 10.1109/access.2021.3052462.
30. **Fehmi, H., Amr, M. F., Bahnasse, A., Talea, M. (2022).** 5G network: analysis and compare 5g NSA/5G SA. Procedia Computer Science, Vol. 203, pp. 594–598. DOI: 10.1016/j.procs.2022.07.085.
31. **LLedó, P. (2017).** Como aprobar el examen de PMP sin morir en el intento. USA: Pablo Lledó.
32. **Mulcahy, R. (2018)**. Preparación para el Examen PMP. Minnetonka.
33. **Salazar-Garces, J. A., Mora-Sánchez, N. V., Romero-Black, W. E., Ollague-Valarezo, J. K. (2020).** Diagnóstico de la aplicación del ciclo PHVA según la ISO 9001: 2015 en la empresa INCARPALM. 593 Digital Publisher CEIT, Vol. 5, No. 6–1, pp. 459–472. DOI: 10.33386/593dp.2020.6-1.440.
34. **Axelos Limited. (2019)**. Courseware ITIL 4 Foundation. IT Preneurs.
35. **Simpson, O., Sun, Y. (2018).** LTE RSRP, RSRQ, RSSNR and local topography profile data for RF propagation planning and network optimization in an urban propagation environment. Data in Brief, Vol. 21, pp. 1724–1737. DOI: 10.1016/j.dib.2018.08.137.
36. **Martínez-Morales, J., Casallas-Hernández, A. (2017).** Proyecto normativo para la reglamentación de la inhibición de señales radioeléctricas en el interior de los

establecimientos penitenciarios. ANE - MinTIC, pp. 1–30.

37. **Hsing-Yi, C., Tsung-Han, L. (2014).** Simulations and measurements of electric fields emitted from a LTE base station in an urban area. International Journal of Antennas and Propagation, Vol. 2014, pp. 1–10. DOI: 10.1155/2014/147341.
38. **Pramono, S., Alvionita, L., Ariyanto, M. D., Sulistyo, M. E. (2020).** Optimization of 4G LTE (long term evolution) network coverage area in sub urban. AIP Conference Proceedings, Vol. 2217, No. 1, pp. 1–9. DOI: 10.1063/5.0000732.
39. **Parra, J. P., Botero, C. (2021).** Pistolas contra Celulares. Fundación Karisma.
40. **Cuevas-Silva, J. M., Mendieta-Izquierdo, G. (2020).** La bioética: retos de la protesta social. Revista Latinoamericana de Bioética, Vol. 19, No. 37–2, pp. 5–7. DOI: 10.18359/rlbi.4706.
41. **Pozo-Leon, E. D. (2022).** El impacto del funcionamiento de los bloqueadores de señales inalámbricas del penal de Ayacucho en la población aledaña en el año 2020. Maestría en Gestión Pública.
42. **Villamizar-Escobar, J. D., Sorzano-Rodríguez, D. M. (2023).** Movilización social ante la transformación de la red móvil en espacios urbanos en el periodo de emergencia sanitaria por COVID-19. Cuadernos de Geografía: Revista Colombiana de Geografía, Vol. 32, No. 2, pp. 366–381. DOI: 10.15446/rcdg.v32n2.107258.

*Article received on 04/06/2024; accepted on 12/09/2024.*
**Corresponding author is Liliana Chanona-Hernández.*

ISSN 2007-9737

# Editorial for Thematic Section:

# New Trends in Artificial Intelligence and its Applications

This thematic section of Computación y Sistemas (CyS) contains fourteen papers presenting advances in Artificial Intelligence (AI) and its applications. AI is a field of study that develops systems and machines capable of performing tasks that typically require human intelligence, such as learning, reasoning, and perception. Since its origin, AI has evolved to incorporate advanced techniques such as deep learning, machine learning, and artificial neural networks. These innovations have enabled significant advancements across various sectors.

The guest editors carefully selected the fourteen papers featured in this thematic issue. At least three scientific committee members evaluated each manuscript in depth. In determining a paper's acceptance, the reviewers considered various aspects, including originality, contribution to the field, soundness, and technical quality.

In the following paragraphs, we have grouped the selected papers by thematic affinity to provide an overview and highlight AI's significant impact on improving efficiency, accuracy, security, and responsiveness across various applications.

**Optimization and Evolutionary Algorithms.** Evolutionary algorithms are a broad field of artificial intelligence in which evolutionary processes inspire algorithms, such as artificial immune systems, inspired by the evolution of acquired immune systems. Combining evolutionary algorithms with optimization creates powerful tools for solving many complex problems, and these algorithms help drive advances in various scientific and engineering disciplines.

An example is the work of Del Angel et al.; the authors propose a methodology for implementing Linear Genetic Programming to evolve programs for the Internet Shopping Multi-item Unit Optimization Problem (ISHOP-U), an NP-Hard optimization problem, propose strategies for readers to quickly implement their proposed approach and produce Linear Genetic Programming algorithms for other problems, giving an example on a test problem. Another work that uses evolutionary algorithms to solve optimization problems is that of Dominguez Guerrero et al.; the authors propose the Normal Attractor Intersection (NAI) and the NAImopso. The NAI avoids the a priori definition of the search direction and equality constraints; it uses a set of attractors covering the entire Pareto front to generate solutions on the Pareto front. The NAImopso is a decomposition-based multi-objective optimization algorithm; we use it to demonstrate the ability of NAI to obtain the Pareto front.

**Artificial Intelligence and Machine Learning.** The articles presented in this section include artificial intelligence techniques to solve various problems; for example, Rodriguez-Arteaga et al. present an architecture to develop a socio-emotional conversational agent that identifies intentions within Socratic dialogues, using a combination of speech act theory and sentiment analysis.

To identify the appropriate intent, this approach considers the influence of personality, the polarity of the words used by the user, and the context in which the dialogue is situated. With this, it is expected to accurately identify communicative intentions, facilitating a natural and empathetic interaction between intelligent virtual agents and users, contributing to the field of artificial intelligence applied to education. Estrada-Patiño et al. use IA to solve climate change problems; they propose HELI, a methodology for temperature forecasting in the context of climate change.

HELI incorporates robust preprocessing that facilitates tuning and learning multiple forecasting strategies, both Classical and machine learning. These strategies are combined and optimized to improve participation and produce a regression function that is at least as effective as the best individually trained method. Some works use learning strategies; for example, in the following works, Srinivas Naveen et al. present a technique

to identify Complex correlations and patterns in data stored in clouds to meet demand.

The proposal offers an innovative approach to decision making since the support systems apply Deep methods Autoencoder Optimization, into the CBA (Cloud-Based Analytics) real. Montalvo-Bereau et al. propose working with spoken language recognition tasks in short sentences, considering two tasks related to speech as auxiliaries in a multitasking architecture. Three models were implemented and trained with different approaches in a single-task and multitask learning paradigm.

The models considered were based on 2D-CNN, one of which was a proposed configuration designed to address less than a second expression. Finally, Mohammed bin Qassim Al-Asiri and Ashwaq Ayed Al-Asmari present an article where they review the state of the art of IA in support of surgical robots; this research highlights the importance of using artificial intelligence robots in surgical operations and increasing awareness of using this technology to increase the success rates of these operations, in addition, the research also shows some gaps in artificial intelligence robot research for surgical operations.

**Cybersecurity and Network Data Analysis.** The selected papers in this thematic section explore various aspects of detecting and preventing cyber threats. Almotairy et al. examine computational propaganda on Twitter, highlighting how Arab propagandists use artificial intelligence to evade detection by constantly changing their writing style and relying on retweet amplification.

The study identifies critical features such as follower ratios and publishing volumes as crucial for detecting propaganda. Eswari and Lakshmi propose an ensemble classifier model to defend against distributed denial-of-service, or DDoS attacks, in traffic flow streams, which overwhelm systems with excessive requests, rendering the system inaccessible.

They incorporate drift detection and streaming properties to analyze service request traffic patterns, showing effectiveness through statistical metrics. Alluraiah and Chetty present the Anomaly-based Real-Time Prevention (ARTP) framework for real-time detection of application-layer DDoS attacks, particularly those targeting web applications. They employ advanced machine learning techniques and validate the framework's efficacy on the LLDoS benchmark dataset.

**Decision Analysis and Knowledge Management.** The selected papers in this section explore various methodologies for effective decision-making and knowledge management. Leyva López et al. present an extension of the ELECTRE III method, using a 2-tuple linguistic representation model to handle heterogeneous information; this allows the decision-maker to adapt preferences to criteria uncertainty.

The proposed method is applied to evaluate the environmental impacts of industrial activities in a petrol station. Olmos-Sánchez examines the need for a comprehensive framework combining Knowledge Management and Systems Thinking to address complex domains. The KMoS-SSA framework is proposed to facilitate the development of intelligent solutions in rapidly evolving contexts. This framework is particularly relevant for transdisciplinary teams working on AI solutions in Complex Informal Structured Domains (CISD), such as work-related stress.

**Classification, Detection, and Data Mining.** The selected papers in this section explore advanced methodologies for object detection and data stream classification. Kokila et al. propose a fully convolutional neural network with a dilation kernel to enhance salient object detection, addressing challenges in multi-level characteristics and distractions.

Their framework outperforms nine cutting-edge saliency detection techniques in accuracy and robustness across various datasets. Althabiti et al. review the state-of-the-art methods in multi-label classification for IoT data streams, which assign multiple class labels to an instance, such as predicting movie genres.

This review focuses on challenges like class imbalance and concept drift. They also highlight gaps for future research in handling the massive, continuous data stream IoT devices generate.

This compendium of works provides an overview of various directions in which AI is developing and its practical applications, highlighting its crucial role in automating complex

ISSN 2007-9737



tasks, improving decision-making, and creating more intelligent and adaptive systems.

Guest editors:

Gilberto Rivera
Laura Cruz-Reyes
Claudia G. Gómez Santillán

Computación y Sistemas, Vol. 28, No. 3, 2024, pp. 1483–1485
doi: 10.13053/CyS-28-3-5202

ISSN 2007-9737

# Spoken Language Identification for Short Utterance with Transfer Learning

Ana Montalvo-Bereau*, Jose Ramón Calvo-de-Lara,
Gabriel Hernández-Sierra, Flavio Reyes-Díaz

Centro de Aplicaciones de Tecnologías de Avanzada,
Cuba

{amontalvo, jcalvo, gsierra, freyes}@cenatav.co.cu

**Abstract.** Spoken language recognition is a research field that has received considerable attention due to its impact on several tasks related to multilingual speech processing. While it has been demonstrated that the use of contextual and auxiliary task information can enhance the results within this field, this avenue has not been fully explored. In the present work, we propose to address the spoken language recognition task in short utterances by considering two speech-related tasks as auxiliaries in a multi-tasking architecture. The primary task was language recognition, with sex and speaker identity serving as auxiliary tasks. Three models from disparate approaches were implemented and trained in a single-task and multi-task learning paradigm. The models considered were 2D-CNN based, one of which was a proposed configuration designed to address less than a second utterances. The experiments were conducted on a subset of the VoxForge corpus, with a markedly limited amount of signals. The results demonstrate that the spoken language recognition task benefits from multi-task learning by using sex and speaker identity as auxiliary tasks over three different models.

**Keywords.** Spoken language recognition, deep learning, transfer learning, multi-task learning.

## 1 Introduction

Spoken language recognition (SLR) is the automated process of identifying the language spoken in a speech sample. It serves as a foundational technology for various multilingual speech-processing applications.

One challenge for voice assistant products is the requirement for predetermined language usage by the user, either explicitly declared or automatically recognized by the system. A reduction in the time required to identify the spoken language in a signal would enhance the user experience, particularly in tasks such as spoken term detection, speech-to-text transcription, and automatic spoken language translation.

Minimizing latency and processing time in spoken language recognition, especially in online mode, is crucial, making the length of the input segment for inference and the size of the model key parameters. At present, SLR systems rely on deep learning models, either in the stage of feature extraction by learning representations [18], or in end-to-end architectures that jointly model feature extraction and classification of the system [2].

Currently, deep neuronal networks with end-to-end architecture lead SLR, especially for short utterances (3 seconds) [23]. Compared with long utterances, the feature representation of short utterances has a large variation, which prevents the model from generalizing well. The challenge of improving the generalization of the model on short utterances remains.

Most machine learning techniques are narrowly focused and trained in isolation with a single task . This approach, which we will refer to as single-task learning, neglects certain fundamental aspects of human learning.

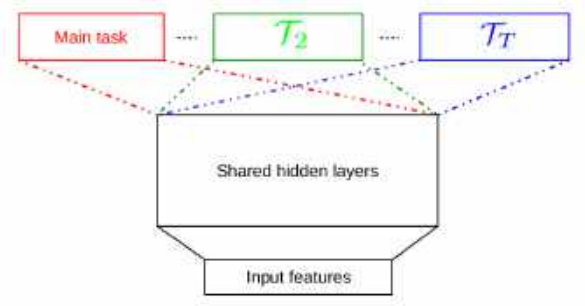



**Fig. 1.** Hard parameter sharing MTL architecture with one main task and $T-1$ auxiliary tasks

Humans enter each new learning task with the knowledge acquired from prior learning experiences. Yet, human learning frequently entails addressing multiple learning tasks concurrently. The set of techniques within machine learning that allows joint learning of related problems is called multi-task learning (MTL [3]).

The basic idea of MTL is to improve the learning of a main task through the use of the information contained in the training signals of other tasks called auxiliary and related to the main one, using a shared representation. This is based on the assumption that what is learned for each task can help other tasks to be learned better. MTL improves generalization by drawing on information contained in related tasks [3].

Speech, as a component of the complex voice signal, conveys semantic information on the message it transmits, but also several non-semantic characteristics, including the speaker's identity, sex, age, language, accent, emotional and health state, and so forth [24].The understanding of these characteristics by automatic learning systems has two main advantages.

First, it could help prevent bias and discrimination in voice applications based on artificial intelligence. Secondly, some studies support the notion that the joint modeling of this information has the potential to positively influence spoken language classification [8]. State-of-the-art approaches, entirely based on deep neural networks, have demonstrated impressive performance for short utterances in SLR [22]. Despite the progress that has been made, these techniques are susceptible to overfitting the training set or domain generalization problems [1].

In this paper, the use of MTL in SLR is evaluated, considering non-semantic tasks, using three different approaches. A preliminary study on this subject was conducted previously [16], and the present work constitutes an extension of that study. The former was the first attempt to assess SLR based on non-semantic multi-task learning, starting from the knowledge transferred from the real images domain to audio classification tasks.

In this study, two additional approaches are considered to test the hypothesis that the inclusion of auxiliary information, such as the identity of the speaker and its sex, could enhance SLR performance for short and very short utterances (less than 3 seconds).

The objective is to experimentally verify that the proposed auxiliary tasks (speaker's identity and sex) for SLR yield benefits of multi-task learning, particularly for very short-duration signals. This is the main contribution of the present work.

The remainder of this paper is organized as follows: section 2 is devoted to a review of the methods required to build a 2D-CNN multi-tasking setup and its applications to SLR. The experimental protocol and dataset used during experimentation are described in section 3, as well as a description of the models assessed. The results are presented and discussed in section 4. Finally, the final conclusions of this work are summarized in 5.

## 2 Methods

The performance metric used is accuracy, defined as the ratio of correctly classified samples (prediction) to the total number of predictions. The value of this metric ranges between 0 and 1 and is expressed as a percentage (%):

$$\text{Accuracy} = \frac{\text{number of correct predictions}}{\text{total number of predictions}}. \quad (1)$$

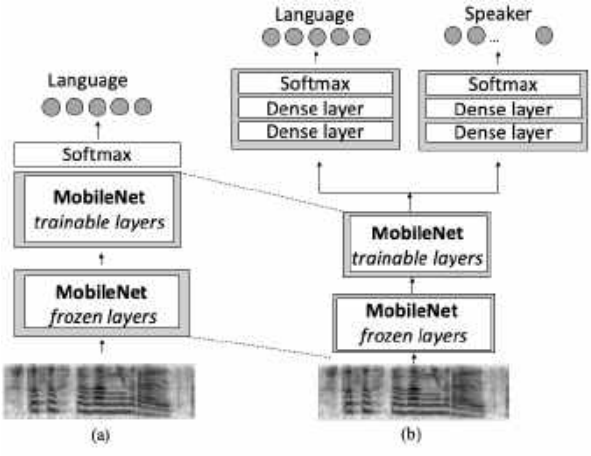



**Fig. 2.** Scheme of MobileNet used in single-task (a) and multi-task (b) setups

### 2.1 Multi-task Learning

Multi-task learning is a collection of techniques intended to learn multiple tasks simultaneously instead of learning them separately [3]. Beyond the biological and pedagogical motivations, from the machine learning perspective, learning multiple related tasks leads to inductive bias, which helps the models generalize better.

This generalization capacity represents a significant aspect when little training data is available and in facing very short-duration test utterances, as addressed in the last model proposed by the authors in the present paper. The concept of multi-task learning (MTL) has been a topic of interest for some time. The rationale behind this approach is that if two related tasks are present, then the learned features should be related as well.

This is particularly relevant in the context of deep models, such as deep convolutional neural networks (CNNs), due to the presence of a hierarchy of features [17].Even if the highest-level features are task-specific, lower-level features can likely be shared. This approach offers the advantage of augmenting the data set for those lower-level features by training them jointly on related tasks.

The two most commonly used methods for performing MTL in deep neural networks are either hard or soft parameter sharing of hidden layers.Hard parameter sharing is the most commonly used [3] and involves sharing the hidden layers between all tasks while maintaining several task-specific output layers. The sharing of representations between related tasks enables the model to generalize better on the main task, thereby reducing the risk of overfitting.

In soft parameter sharing, each task has its own model, and the learning process penalizes the distance between the different parameters. Unlike hard sharing, this approach provides greater flexibility for the tasks by only loosely coupling the shared space representations.

In situations where it is necessary to solve several classification tasks simultaneously, MTL is an optimal solution. However, in most situations, the focus is on the performance of a single task.

The relationship between tasks is a crucial aspect of MTL, although it is challenging to assess [29]. The application of MTL for SLR has primarily focused on relating the phonetic information with the language, either in end-to-end approaches [10] as during the representation stage [29].

Additionally, there are instances where the recognition of language or dialect serves as an auxiliary task, with the primary objective being to relate it to phonemes to enhance the efficacy of speech recognition [13].

In the case of negatively correlated tasks, such as language and domain differences, adversarial MTL has been employed to develop a model that is less reliant on the domain [1]. Some theoretical advances have been made in understanding task-relatedness ; however, there has been limited progress towards this goal.

To compensate for this, researchers have recently explored MTL from a more experimental point of view, correlating performance gains with task properties to achieve a better understanding of when models can profit from auxiliary tasks [10, 15].

This paper sheds light on the specific task relations that can lead to gains from MTL models over single-task setups in SLR. To the best of the authors' knowledge, the consideration of speakers'

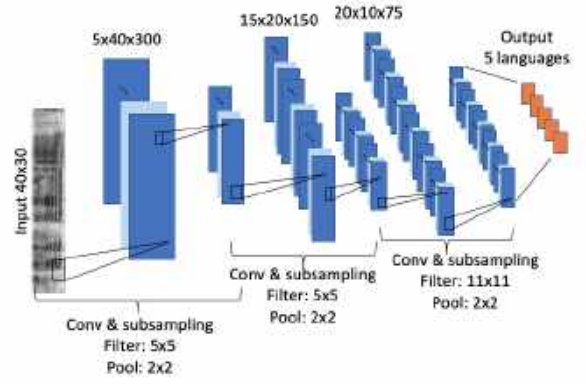



**Fig. 3.** Representation of Lozano's single-task model architecture, with 3 hidden layers of 5, 15 and 20 filters respectively to discriminate among 5 languages

identity and sex as auxiliary tasks for spoken language recognition in a multi-tasking setup has not been explored before.

### 2.1.1 Auxiliary Tasks

It is not reasonable to assume that information gathered through the learning of a set of tasks will be relevant to the learning of another task that has nothing in common with the already learned set of tasks. From an engineering perspective, speech recognition and speaker recognition are independent tasks.

However, the human brain interprets and decodes information from both speaker traits and linguistic content from speech in a joint corroborative manner [8]. Similarly, unified frameworks for speaker and language recognition have been attempted using a shared deep neural network, which outperforms the single-task implementation.

On the one hand, language and speaker recognition tasks share numerous common techniques, including cepstral feature extraction and well-established Gaussian-based modeling. On the other hand, researchers in both areas have a history of learning from each other. For instance, the success of i-vector [5] and x-vector [26] representations originally proposed for speaker recognition has been immediately transposed to language recognition [25]. Several technologies are shared between speakers and language recognition. Consequently, the proposed ideas in one application can be also used in another. In [4] speaker identity and sex have been demonstrated to be correlated, thereby establishing a link between these non-semantic tasks and the identity of the speaker.

As for selecting which auxiliary tasks to employ, these studies have strongly encouraged us to explore the benefits of using sex and speaker identity to the SLR main task. The rationale for employing both auxiliary tasks is to direct the networks' attention to the correlation between the variability of language posteriors and two of the speaker attributes. If the system can differentiate the speaker's characteristics, then this information can be utilized for a more accurate interpretation of the distortion introduced by one speaker in comparison to another.

### 2.1.2 Multi-task Architecture

The multi-task architecture employs hard parameter sharing, a technique originally proposed by [3] that has remained the norm for almost 30 years. Figure 1 represents a hard parameter sharing MTL model, with one main task and $T-1$ auxiliary tasks, where $T$ is the number of tasks. There are two fundamental aspects shared between these MTL systems.

First, all tasks use the same base representation of the input data. Second, the task-specific portions of the network all begin with the same representation from the final shared layer, as all the tasks share the same network weights until bifurcation. In this setup, each task contributes to the cost function with its own individual loss, as illustrated in equation 2.

For the sake of simplicity, lets consider $T$ tasks, denoted as $\mathcal{T}_1, \mathcal{T}_2 ... \mathcal{T}_T$. The training data of each task is represented as $\mathcal{D}^{\mathcal{T}_t}$ where $t \in \{1, 2, ..., T\}$. Being $\mathcal{L}_{\mathcal{T}_t}(\mathcal{D}^{\mathcal{T}_t}, \theta)$ the loss function of the task $\mathcal{T}_t$ and $\theta$ the total parameters of the MTL model, the objective is to estimate the model parameters $\theta^*$ such that:

$$\theta^* = \arg\min_{\theta} \sum_{t=1}^{T} \lambda^{\mathcal{T}_t} \mathcal{L}_t(\mathcal{D}^{\mathcal{T}_t}, \theta), \qquad (2)$$

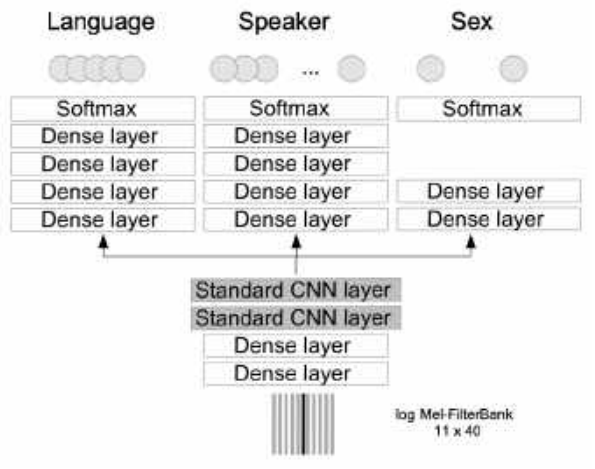



**Fig. 4.** Representation of multi-task frame level model

where $\lambda^{\mathcal{T}_t}$ is the non-negative weight, which considers the impact the particular task $\mathcal{T}_t$ will have on the estimation of the system parameters.

In most cases, the auxiliary tasks are dropped at test time, keeping only the main task outputs. For single-task setup, only the main task remains.

### 2.2 2D-CNN

The three models assessed are CNN-based. Deep convolutional neural networks (CNNs) have been demonstrated to be effective in reducing spectral variations and modeling correlations in acoustic features [28], and have been utilized in a multitude of audio classification tasks [9]. There exist CNN-based approaches that utilize raw or minimally preprocessed audio as an input, employing one-dimensional convolutions. However, the majority of outcomes have been achieved through the utilization of two-dimensional (2D) CNNs on spectrograms [19].

In general, deep CNN models comprise multiple convolution layers connected to one or several fully connected layers. The convolution layers may be regarded as feature extraction layers, while the fully connected layers may be considered as classification layers. To feed the 2D-CNN-based models, acoustic features were extracted from the audio signals and presented in the time domain as spectrograms.

The acoustic features were computed using Kaldi [20]. The signal is subjected to a pre-emphasis filter, then divided into 20 millisecond frames (with 10 ms overlap), and a window function is applied to each frame. Subsequently, a short-time Fourier transform is computed to obtain the power spectrum.

Subsequently, a triangular Mel scale filter bank and the logarithm of the energy output of the individual bandpass filter are applied, resulting in a representation of 40-dimensional vectors known as a spectrogram or log Mel filter bank (log Mel-FBank). MFCCs [6] are the most widely used input features for speech analysis tasks.

To obtain MFCCs, a discrete cosine transform is applied to the log Mel-FBank, retaining a number of the resulting coefficients while discarding the rest. It has been found that the last step removes information and destroys spatial relations; therefore, it is usually omitted, which yields the log Mel-FBank output, a popular feature across the speech community.

## 3 Experimental Setup

### 3.1 Corpus

The corpus utilized was VoxForge [12], a free and open-source corpus of voices containing samples of more than 18 different languages. The data consisted of audio files of approximately 5 to 10 seconds in duration, accompanied by the transcription of the spoken text, as well as labels related to the language, sex, and identity of the speaker.

As the corpus is comprised of audio samples submitted by individuals from diverse geographical and linguistic backgrounds, the quality of the samples varies according to the recording conditions and equipment used by the contributors.

This results in a significant variation of speech quality between samples, which is representative of real-world scenarios. For experimentation, we defined a VoxForge subset consisting of five languages: German, Spanish, French, English, and Russian. Approximately 38 minutes per language were allocated for the creation of training, validation, and test sets, ensuring a balance of

**Table 1.** Language accuracy comparison of single and multi-task architectures using MobileNet base model

| Setup | Acc train (%) | Acc val (%) | Acc test (%) |
|---|---|---|---|
| STL | 100 | 80.9 | 76.5 |
| MTL | 99.7 | 82.86 | 83.42 |

**Table 2.** Language accuracy comparison of single and multi-task architectures using Lozano's model adaptation

| Setup | Acc train (%) | Acc val (%) | Acc test (%) |
|---|---|---|---|
| STL | 96.67 | 93.92 | 91.39 |
| MTL | 98.24 | 97.65 | 96.44 |

female and male speakers. Although this is not an objective pursued in this research, the defined scenario has a very low data-resources profile compared to other speech-related applications.

### 3.2 Models Description

The influence of MTL on spoken language recognition across three distinct models is evaluated. The main task for all models is language recognition. The first model introduces speaker identity as auxiliary task, while the next two models employ both speaker and sex identification tasks as auxiliaries.

Whereas the initial two models (MobileNet pretrained model and Lozano's model) utilize short utterances (3-second duration samples), the proposed third model employs very short duration utterances (110 ms). The parameter tuning for each of the investigated methods was conducted over the validation set.

#### 3.2.1 MobileNet Pre-trained Model

The first SLR approach evaluated is an end-to-end architecture based on a CNN pre-trained on a set of images [21], for a more detailed explanation, please refer to previous work [16]. This technique of transferring knowledge is known as transfer learning. In this case, the parameters of the initial network are trained on images of real objects that are very different from the spectrograms that constitute the input to this method.

As input, we considered the use of spectrograms of the audio signal. This two-dimensional representation in the time-frequency domain can be considered as an image. The 300 initial temporal vectors from the spectrogram of each signal were concatenated to form a $40 \times 300$ feature matrix, which was subsequently transformed into a gray-scale image. Only the first three seconds of audio from each signal were utilized after the initial silence segments were eliminated.

To initialize the network, the MobileNetV2 [21] was employed as a pre-trained model. MobileNetV2 was developed by Google and trained on the Imagenet data set, which comprises 1.4 million images and 1,000 types of web images. The proposed model comprises the MobileNetV2 with 53 layers of depth, of which the initial 30 remain fixed or "frozen" and the remaining network layers are re-trained (see Figure 2). The initial block will be referred to as pre-trained and can be considered a high-level abstraction features extractor, which transforms the input sequence into a characteristics map.

The pre-trained block is succeeded by trainable layers of the convolutional block, which are also part of the MobileNetV2. However, unlike the pre-trained block, their parameters are learned over the training set. In the multi-task setup, with MobileNet's trainable layers, two individual task branches are trained: language as the main task and speaker identity as the auxiliary task [16].

The layers of the individual branches are dense layers with 512 neurons, using the ReLU activation function and a 25% dropout. The Adam optimizer algorithm was employed, as well as a learning rate of $10^{-4}$. It is important to note that the estimation of the model parameters is directly affected by the knowledge embedded in the frozen layers, which is transferred to the trainable layers that follow it.

#### 3.2.2 Lozano's Model

The model proposed by [11] was one of the first attempts to build an SLR system intended to deal with short utterances using CNNs as end-to-end approach. In the referenced paper, the network was fed with speech segments of three seconds, in

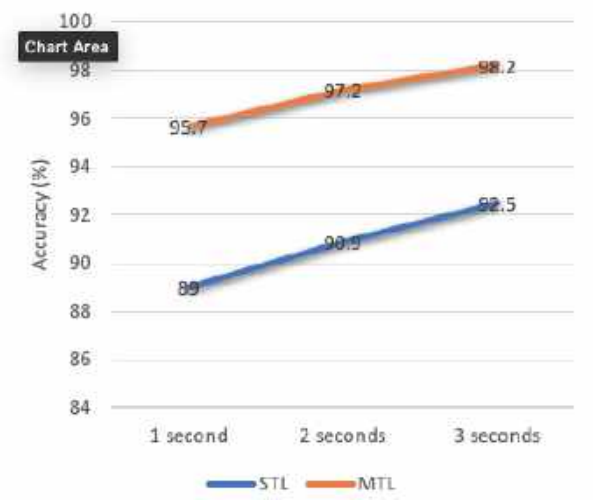



**Fig. 5.** Accuracy with posteriors combination of frame-level model using majority voting for language recognition task over test set

the form of a matrix of dimensions $56 \times 300$, formed by 300 frames.Each frame is represented with a vector of 56 Mel Frequency Cepstral Coefficients and its derivatives (MFCC-SDC) [27]. The CNN system proposed then, obtained comparable results to the i-vector approach, having much less free parameters.

The CNN system proposed then obtained comparable results to the i-vector approach, having much less free parameters. We have replicated Lozano's network architecture, modifying the dimensions of the input and output layers (see Figure 3 for the modified version). This results in a more streamlined configuration than the MobileNet pre-trained model in terms of the number of parameters to estimate, although it still faces the challenge of recognizing speech signals of at least three seconds in duration.

In the adaptation of Lozano's model presented in this work, instead of MFCC-SDC acoustic features, log Mel-FBank features composing the spectrogram were utilized. It has been demonstrated that convolution on mel spectrograms is more beneficial than on decorrelated coefficients [14].

Each speech signal was segmented into intervals of 3 seconds, with an overlap of 50%. The aforementioned segments were represented by a matrix of dimensions $40 \times 300$ formed with the log Mel-FBank features and were used to feed the 2D-CNN models.

Details about STL Lozano's model can be seen in Figure 3, where 3 hidden layers of 5, 15, and 20 filters define the architecture as proposed in the original paper. In the case of the multi-task setup, the output of the third convolutional layer was flattened and passed on to the three branches in parallel.

The language, speaker, and sex branches are formed with two dense layers of 256 and 128 neurons, respectively, a dropout factor of 25%, and a softmax activation function at the end. Regarding the training of the network, the algorithm used is stochastic gradient descent with a learning rate of 0.01 and based on minibatches of 32 samples.

### 3.2.3 Frame-level Model

The SLR frame-level approach, which was initially proposed in [7], demonstrates how deep neural networks are particularly well-suited for SLR in real-time applications. This is due to their capacity to emit a language identification posterior at each new frame of the test utterance. In [7], the authors compare their proposal with the i-vector approach using very short test utterances ($\leq 3s$).

In contrast, our study aims to investigate the potential contribution of MTL to the system under similar test duration conditions. Our proposal focuses on SLR with very short utterances, less than 3 seconds speech segments. Each frame is represented with its log Mel-FBank feature vector, and spliced together with 5 left and 5 right context features to form a $40 \times 11$ dimensional feature matrix.

The spliced features are fed as input to the 2D-CNN, thus enabling the model to be trained with samples corresponding to a temporal context of $11 \times 0.01 = 0.11$ seconds. The spliced features are fed as input to the 2D-CNN, so the model is trained with samples corresponding to a context of $11 \times 0.01 = 0.11$ seconds.

**Table 3.** Language accuracy comparison of single and multi-task architectures using frame-level model

| Setup | Acc train (%) | Acc val (%) | Acc test (%) |
|---|---|---|---|
| STL | 95.39 | 79.08 | 78.35 |
| MTL | 97 | 88.82 | 88.11 |

**Table 4.** SLR multi-task models' relative improvement over test set. Accuracy(%) reported

| Model | STL | MTL | Improvement |
|---|---|---|---|
| Mobile | 76.50 | 83.42 | 6.92 |
| Lozano | 91.39 | 96.44 | 5.05 |
| Frame level | 92.50 | 98.20 | 5.70 |

As illustrated in Figure 4, the lower-level representations of the network across the tasks are shared. The first pair of layers is fully connected with 256 neurons, followed by a convolutional layer comprising a convolution and max-pooling operation. The convolution operation is achieved by weight sharing across the entire training sample. A total of 256 filters and ReLU activations are employed, with each 2D convolution operated with zero-padding using a $3 \times 3$ kernel size.

A second convolutional layer is utilized to model the local spectro-temporal correlations of the speech spectrogram, this time using 128 filters of the same size. The resulting feature map is then flattened and passed on to dense layers for language, speaker, and sex mapping, as illustrated in Figure 4 of the MTL setup.

In the MTL setup, the individual branches comprise four dense layers for language and speaker classification and two dense layers for the binary task of sex classification. The STL setup employs a configuration similar to Figure 4 which is designed to ignore the speaker and sex branches.

During single-task estimation, the output is the classification label corresponding to the most probable language. Due to the limited size of the dataset, the network is susceptible to overfitting. To address this issue, we employ dropout training (with a factor of 0.25) in the feedforward network.

## 4 Results and Analysis

The results and analysis section presents the findings of the experiments conducted. The three models were trained and evaluated in single and multi-task setups using the same data sets. The results are presented on the training, validation, and testing sets. The performance of the models was assessed using the accuracy metric formulated in equation 1.

Table 1 presents the results of the MobileNet base model. The first row of the table exhibits the accuracy achieved by the single-task model, while the second row contains the accuracy of the multi-task model trained using speaker identity as an auxiliary task. The training process of the single and multi-task models was stopped at epoch 20, as the accuracy reached a plateau at that value.

From Table 1 it can be seen that the language accuracy value on the test set for the multi-task architecture is higher than in the single-task. This could be indicative of the beneficial effect of incorporating speaker identity information into the proposed architecture, which enhances the model's discriminatory capacity in the SLR task.

It is also noteworthy that the discrepancy in language accuracy between the validation and testing sets diminishes as the single-task model transitions to the multi-task architecture, suggesting greater generalizability in the latter. In the case of Lozano's model adaptation, a second auxiliary task (speaker's sex) was included, and the same behavior was observed in Table 2.

This table demonstrates that multi-task outperforms single-task and that the gap between validation and test accuracy is smaller for the multi-task model. Table 3 presents the results of our frame-by-frame SLR approach. As anticipated, the performance of the SLR is degraded in comparison to the Mobile and Lozano models, given the 30-fold reduction in signal evaluation time (0.11s vs. 3s).

However, at the frame level, the speaker's identity and sex contribution to the SLR is well received by the multi-task model, which consistently improves performance. A fair comparison between the frame-level model and those based on three-second spectrograms can

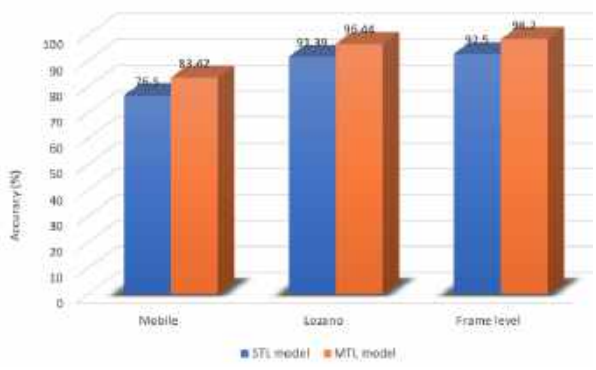



**Fig. 6.** SLR multi-task models' accuracy(%)

be made by deriving decisions about language identification at the utterance level by combining frame posteriors. A common and simple approach used in the literature is majority voting, where, at each frame, the language associated with the highest posterior receives a single vote while the rest receive none. The voting scheme aims to control the negative effect of outlier scores. The score for a given language $l$, $s_l$, is then computed by counting the received votes over all the $N$ frames as follows:

$$s_l = \sum_{t=1}^{N} \delta(p(L_l|x_t, \theta)), \tag{3}$$

with $\delta$ function defined as:

$$\delta(p(L_l|x_t, \theta)) = \begin{cases} 1, & \text{if } l = \arg\max_l(p(L_l|x_t, \theta)), \\ 0, & \text{otherwise.} \end{cases}$$

where $p(L_l|x_t; \theta)$ represents the class probability output for the language $l$ corresponding to the input example $x_t$ at time $t$, by using the model defined by parameters $\theta$.

Figure 5 collects the results of the majority voting over 1, 2, and 3-second intervals for the frame-level model. It is worth noting that for equal test utterance durations (3s), the frame-level approach performs better than the approaches of Mobile and Lozano. The posterior combination also demonstrates the improvement of the MTL approach in comparison to the STL.

The combination of language posteriors places frame-level model in a superior position in terms of performance, as shown in Figure 6. However, when considering the improvement of MTL, as shown in Table 4, the three models are quite similar, with a slight advantage for the MobileNet-based model.

## 5 Conclusions

All of the evaluations demonstrated the superiority of multitask learning for SLR when using speaker-related non-semantic characteristics, such as identity and sex, as auxiliary tasks. Initially, the hypothesis was verified for signals of three seconds, with and without image-based pretraining. Subsequently, an approach to deal with very short-duration utterances was proposed, and for both frame and utterance levels, MTL resulted in a superior SLR performance.

This article makes two main contributions: first, it determines the auxiliary tasks that should be used in a multi-task approach to SLR; second, it verifies that this approach will be beneficial. Additionally, it proposes a frame-level model with a convolutional neural network (CNN), which is a proposal much closer to real-time applications. The study and deepening of MTL for SLR offers possibilities for its use in low data resources environments.

## References

1. **Abdullah, B. M., Avgustinova, T., Möbius, B., Klakow, D. (2020).** Cross-domain adaptation of spoken language identification for related languages: The curious case of slavic languages. Proceedings of the Interspeech 2020, pp. 477–481. DOI: 10.21437/Interspeech.2020-2930.
2. **Cai, W., Cai, D., Huang, S., Li, M. (2019).** Utterance-level end-to-end language identification using attention-based CNN-BLSTM. Proceedings of the ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and

Signal Processing, pp. 5991–5995. DOI: 10.1109/ICASSP.2019.8682386.

3. **Caruana, R. (1993).** Multitask learning: a knowledge-based source of inductive bias. Proceedings of the Tenth International Conference on International Conference on Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 41–48. DOI: https://doi.org/10.1016/B978-1-55860-307-3.50012-5.

4. **Chen, H., Xu, L., Yang, Z. (2018).** Multi-dimensional speaker information recognition with multi-task neural network. Proceedings of the IEEE 4th International Conference on Computer and Communications, pp. 2064–2068. DOI: 10.1109/CompComm.2018.8780705.

5. **Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., Ouellet, P. (2011).** Front-end factor analysis for speaker verification. Proceedings of the IEEE Transactions on Audio, Speech, and Language Processing, Vol. 19, No. 4, pp. 788–798. DOI: 10.1109/TASL.2010.2064307.

6. **Furui, S. (1986).** Speaker-independent isolated word recognition based on emphasized spectral dynamics. Proceedings of the ICASSP'86. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 11, pp. 1991–1994. DOI: 10.1109/ICASSP.1986.1168654.

7. **Gonzalez-Dominguez, J., Lopez-Moreno, I., Moreno, P. J., Gonzalez-Rodriguez, J. (2015).** Frame-by-frame language identification in short utterances using deep neural networks. Neural Networks, Vol. 64, pp. 49–58. DOI: 10.1016/j.neunet.2014.08.006.

8. **Kumar, R., Yeruva, V., Ganapathy, S. (2018).** On convolutional LSTM modeling for joint wake-word detection and text dependent speaker verification. Interspeech, pp. 1121–1125. DOI: 10.21437/Interspeech.2018-1759.

9. **Latif, S., Rana, R., Khalifa, S., Jurdak, R., Epps, J. (2019).** Direct modelling of speech emotion from raw speech. DOI: 10.48550/ARXIV.1904.03833.

10. **Li, Z., Zhao, M., Li, J., Zhi, Y., Li, L., Hong, Q. (2020).** The XMUSPEECH system for the AP19-OLR challenge. Proceedings of the Interspeech 2020, pp. 452–456. DOI: 10.21437/Interspeech.2020-1923.

11. **Lozano-Diez, A., Zazo-Candil, R., Gonzalez-Dominguez, J., Toledano, D. T., González-Rodríguez, J. (2015).** An end-to-end approach to language identification in short utterances using convolutional neural networks. Proceedings of the INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, ISCA, pp. 403–407. DOI: 10.21437/INTERSPEECH.2015-164.

12. **MacLean, K. (2009).** Voxforge. http://www.voxforge.org/.

13. **Mendes, C., Abad, A., Neto, J. P., Trancoso, I. (2019).** Recognition of latin american spanish using multi-task learning. Proceedings of the Interspeech 2019, pp. 2135–2139. DOI: 10.21437/Interspeech.2019-2772.

14. **Mohamed, A. (2014).** Deep neural network acoustic models for ASR. Ph.D. thesis, University of Toronto, Canada.

15. **Montalvo, A., Calvo, J. R., Bonastre, J. F. (2020).** Multi-task learning for voice related recognition tasks. Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, ISCA, pp. 2997–3001. DOI: 10.21437/INTERSPEECH.2020-1857.

16. **Montalvo-Bereau, A., Reyes-Díaz, F., Hernández-Sierra, G., Calvo-de-Lara, J. R. (2022).** Identificación de idioma hablado en señales cortas aplicando transferencia de aprendizaje. Revista Cubana de Ciencias Informáticas, Vol. 16, No. 1, pp. 77–91.

17. **Mukherjee, S., Shivam, N., Gangwal, A., Khaitan, L., Das, A. J. (2019).**

Spoken language recognition using CNN. Proceedings of the International Conference on Information Technology, pp. 37–41. DOI: 10.1109/ICIT48102.2019.00013.

18. **Padi, B., Ramoji, S., Yeruva, V., Kumar, S., Ganapathy, S. (2018).** The LEAP language recognition system for LRE 2017 challenge - improvements and error analysis. Proceedings of the Odyssey 2018: The Speaker and Language Recognition Workshop, pp. 31–38. DOI: 10.21437/ODYSSEY.2018-5.
19. **Palanisamy, K., Singhania, D., Yao, A. (2020).** Rethinking CNN models for audio classification. DOI: 10.48550/arXiv.2007.11154.
20. **Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., Vesely, K. (2011).** The Kaldi speech recognition toolkit. Proceedings of the IEEE 2011 workshop on automatic speech recognition and understanding.
21. **Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., Chen, L. (2018).** MobileNetV2: Inverted residuals and linear bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.
22. **Shen, P., Lu, X., Kawai, H. (2022).** Transducer-based language embedding for spoken language identification. Interspeech, pp. 3724–3728. DOI: 10.48550/ARXIV.2204.03888.
23. **Shon, S., Ali, A., Glass, J. R. (2018).** Convolutional neural networks and language embeddings for end-to-end dialect recognition. DOI: 10.48550/arXiv.1803.04567.
24. **Shor, J., Jansen, A., Maor, R., Lang, O., Tuval, O., Quitry, F. d. C., Tagliasacchi, M., Shavitt, I., Emanuel, D., Haviv, Y. (2020).** Towards learning a universal non-semantic representation of speech. Interspeech 2020. DOI: 10.21437/interspeech.2020-1242.
25. **Snyder, D., Garcia-Romero, D., McCree, A., Sell, G., Povey, D., Khudanpur, S. (2018).** Spoken language recognition using x-vectors. Odyssey 2018: The Speaker and Language Recognition Workshop, pp. 105–111. DOI: 10.21437/ODYSSEY.2018-15.
26. **Snyder, D., Ghahremani, P., Povey, D., Garcia-Romero, D., Carmiel, Y., Khudanpur, S. (2016).** Deep neural network-based speaker embeddings for end-to-end speaker verification. Proceedings of the IEEE Spoken Language Technology Workshop, IEEE, pp. 165–170. DOI: 10.1109/SLT.2016.7846260.
27. **Torres-Carrasquillo, P. A., Singer, E., Kohler, M. A., Greene, R. J., Reynolds, D. A., Deller-Jr., J. R. (2002).** Approaches to language identification using Gaussian mixture models and shifted delta cepstral features. Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002), pp. 89–92. DOI: 10.21437/ICSLP.2002-74.
28. **Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Laurent, C., Bengio, Y., Courville, A. C. (2017).** Towards end-to-end speech recognition with deep convolutional neural networks. DOI: 10.48550/arXiv.1701.02720Focustolearnmore.
29. **Zhao, M., Li, R., Yan, S., Li, Z., Lu, H., Xia, S., Hong, Q., Li, L. (2019).** Phone-aware multi-task learning and length expanding for short-duration language recognition. 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, pp. 433–437. DOI: 10.1109/APSIPAASC47483.2019.9023014.

*Article received on 28/02/2024; accepted on 14/05/2024.*
**Corresponding author is Ana Montalvo Bereau.*

ISSN 2007-9737

# Role of Artificial Intelligence in Supporting the Performance of Surgical Robots: A Survey

Mohammed bin Qassim Al-Asiri*, Ashwaq Ayed Al-Asmari

King Abdulaziz University,
Saudi Arabia

eng_911@hotmail.com, ashwaqalasmry99@gmail.com

**Abstract.** The applications of AI in medicine go well beyond what people may think. With an application that analyses data and forecasts infections and diseases, diseases can be predicted. In addition to giving physicians the ability to quickly and accurately assess thousands of patients, this aids in their decision-making process. As a result, doctors can eliminate the lengthy time it takes to analyze these tests without diminishing their importance because they will continue to be an integral part of doctors' medical routes, serving as an auxiliary tool. When AI is used correctly, diseases can be detected early, thereby reducing government expenditures, as this area carries significant opportunities for the growth of the healthcare sector.This study aims to shed light on the latest developments regarding the reality of using artificial intelligence and surgical robots in the healthcare sector and whether they are compatible with the wise leadership's vision for the transformation of the healthcare sector in the Kingdom of Saudi Arabia, which aims to restructure the healthcare sector as it is a fundamental pillar of the Kingdom's Vision 2030. Are we keeping up with this technological development in the healthcare sector in Saudi Arabia, and where are we heading? What is the future of artificial intelligence and surgical robots? Will robots replace human surgeons? This research relied on a survey critique methodology, reviewing previous studies between 2011 and 2021 on artificial intelligence and surgical robots. The study reached several conclusions, including the following: There is a level of responsibility and patients' ability to accept surgical robots. The clinical medical trend is to increase reliance on the capabilities of artificial intelligence and surgical robots in predicting diseases and therapeutic diagnosis. The reliance on artificial intelligence in processing big data and its development in fields such as radiology, imaging, and data analysis.

**Keywords.** Artificial intelligence, robotic surgeries, medical systems, automation, endoscopy, robots, Internet of Things (IoT), ophthalmology, spinal surgery, prostate surgery, machine learning, deep learning, and blockchain.

## 1 Introduction

The healthcare sector in Saudi Arabia has achieved many accomplishments in the previous stage, such as improving the quality and efficiency of healthcare services and facilitating their access through the digitization of the healthcare sector, launching a package of applications (Sehaty, Mawid) and increasing services to cover all regions of the Kingdom.

This highlights the importance of artificial intelligence in developing the healthcare sector and the opportunity for rapid improvement in healthcare quality [1]. This requires the increasing role of artificial intelligence in the field of medicine in the future. In a report prepared by Accenture Consulting in 2014, it was found that the global market value of artificial intelligence in medicine reached $600 million. By 2021, the indications showed that this number had reached $6.6 billion [2].

The focus of this research is on predicting what a visit to the doctor will be like in the next 20 years in light of the astonishing and rapid developments in the field of communications and information technology, including the use of artificial intelligence and surgical robots as supportive tools to begin building a future of appropriate healthcare for both the surgeon and the patient.

This research highlights the importance of using artificial intelligence robots in surgical operations and increasing awareness of using this technology to increase the success rates of these

operations. The research also shows some gaps in artificial intelligence robot research for surgical operations due to their low usage worldwide, despite their many benefits, which reduce complications and the duration of surgical operations [3].

Based on the above, the problem of this research revolves around answering the following question: How did the science of artificial intelligence begin and develop in the medical field, and what is its vital role in activating the performance of surgical robots?

## 2 Surgical Robots and their Multiple Uses

The National Aeronautics and Space Administration (NASA) of the United States investigated the idea of remote surgery in the 1970s,intending to use it on astronauts in orbit. The main concept was to create a surgical tool-equipped machine that could be operated by a ground-based surgeon and installed in a space station. The removal of undesired movement was the other objective of robotic surgery.

The first surgical robot was utilized in a laparoscopic procedure in 1985. An Imperial College London-developed robot was used for a prostate operation in 1988. In 1992, a robot developed by Integrated Surgical Systems was used to prepare a cavity in the thigh bone and replace the hip joint, and the task was successfully executed with greater precision and speed than human surgeons [4].

The 1980s and 1990s were known as the laparoscopic surgery era. A laparoscope is a thin, lighted optical device that looks like a small telescope and inspects the abdominal and thoracic chambers. Surgeons discovered they could operate on patients with smaller incisions, shortening their hospital stay and recovery period.

Since this method is one kind of minimally invasive surgery, some surgeons have worked to create surgical robots that can help with this kind of surgery. In the field of artificial intelligence, robots are the most exciting devices. While it may seem premature to discuss the possibility of robots performing surgery in the far future, imagination has become a reality.

A Chinese robot that used artificial intelligence to pass the nation's professional exam did it in 2017. The surgical robot Versius, created by the British medical technology company CMR Surgical, carried out its first accurate surgical procedure in colon and rectal surgery in February 2022. Versius has many arms that can help surgeons in operating rooms [5].

Even in the most developed nations, the congestion of hospitals and medical centres and the pressures placed on physicians and nurses are persistent issues facing the healthcare industry. For instance, a 2016 study done in the United States found that 96% of patient complaints centre around difficulty completing registration forms and unsatisfactory encounters at reception desks.

As a result, several artificial intelligence-based systems and apps have attempted to address these issues. This was supported by a study [6], which stated that the pandemic posed a major challenge to healthcare workers and hospitals and that robots and artificial intelligence could be used to reduce communicable contamination and assist patients and management in the surgical environment during times of high patient flow to healthcare centres and hospitals.

To ascertain whether device diagnoses were better and more accurate than physician diagnoses, researchers at Northwestern University in Illinois, USA, collaborated with Google and numerous medical centres in 2019. The researchers examined data used with the consent of thousands of cancer patients.

Although artificial intelligence in healthcare is still in its early stages of research and is not yet suitable for general application, a recent study featured in The New York Times and published in the journal "Nature Medicine" offers some insight into what lies ahead for the technology.

Medical robots assist surgeons in performing precise and complex surgical operations, where the surgeon controls the robot's arms remotely, and they can reach precise locations that the human hand cannot reach. The camera provides the surgeon with highly precise vision of the surgical site.

The advantages of using robots in surgical operations include precision, flexibility, and reduced complications resulting from surgical operations, such as infections, bleeding, and

reduced vibrations. This was confirmed by a study [7] that showed that there are four main areas where artificial intelligence methodology is used in medicine: diagnosis, predictive methods, robotic surgery, and therapeutic diagnosis, where robotic oral surgery provides less risky treatment methods and better results in terms of reducing blood loss and quick recovery.

At an advanced level, robots can now assist surgeons in many minimally invasive surgical procedures. For example, in operations to remove a tumour that has affected one of the body's organs, the doctor cannot be 100% certain of the exact location of the tumour, the actual distance of the arteries adjacent to the tumour, or the possibility of the surgical scalpel injuring adjacent organs.

In addition, robots and computers can provide a great service to surgeons. After computed tomography (CT) imaging, the tumour's location can be accurately determined, and the distance of adjacent organs can be measured. Additionally, a three-dimensional image can be formed to assist the doctor during the surgical procedure.

A study [8] confirmed that gastroenterology is an area artificial intelligence can significantly impact. It is possible to analyze images, perform accurate evaluations, and provide more information than traditional analysis, helping surgeons in gastrointestinal surgical procedures.

The surgeon controls the robot's arms by placing a control device in front of a computer screen in the assigned spot. It would be truer to state that the robot programmes the motions the surgeon wishes to make and then executes those actions on the patient's body. The constraints of surgical intervention have been greatly increased by robot-assisted surgery, which outperforms open surgery in several ways, including increased accuracy, fewer incisions, reduced pain, and decreased blood loss.

The surgeon makes a tiny, barely noticeable incision in the patient's abdomen, inserts a tiny robotic arm with a camera attached to it via a computer screen, and the arm moves through the patient's organs based on the surgeon's observations made in front of the computer screen, which is magnified ten times. The application of artificial intelligence in robotic surgery has been shown in a study to substantially impact future training and improve the surgical experience during the process to raise the standard of surgical care.

## 3 Virtual Healthcare and Cloud Computing

Using technology like video communication, messaging, or software, virtual healthcare, commonly called remote healthcare, facilitates quicker contact between patients and doctors located far away. Many patients use fitness trackers or health monitoring gadgets to monitor any changes in their health and communicate that information to their physicians.

With the capacity to do tests and quickly monitor their health from home, this technology allows patients to be completely comfortable and facilitates speedy communication between them and their doctor. Furthermore, it lessens the need for travel by enabling patients to consult with professionals outside of the city. These days, many web-based or mobile applications are available for obtaining information, including ones that read blood glucose levels, measure blood pressure, measure blood glucose levels, or measure the function of the lungs.

In this sense, the primary care physician forwards to the specialist for review of examination notes, medical history, test results, X-rays, or other imaging tests. The specialist may ask to meet with the patient, arrange a virtual appointment at the doctor's office, or reply electronically. These virtual consultations can eliminate the need for pointless in-person referrals to the specialist and shorten the time needed to hear back from the doctor [9].

This was addressed in a study [10], which explained that artificial intelligence applications and machine learning have helped support diagnostic decision-making and dental treatment with the help of data-driven algorithms and called for the use of advanced modern technologies such as Dentronics to help improve healthcare.

Cloud computing is an important technology that provides the ability to store, process, and analyze data necessary for the application and operation of artificial intelligence and machine learning systems and to drive innovation. The AWS cloud platform facilitates the absorption and

processing of data, whether structured, unstructured, or continuously streaming, and simplifies the process of building, training, and deploying machine learning models.

With this technology, healthcare facilities worldwide can share information on genetic medical analyses while maintaining patient data [11]. However, a study [12] noted that artificial intelligence is expected to face significant challenges in independent development, data mining, and machine learning.

## 4 Surgical Robots

Pattern recognition and image interpretation are skills humans use to read microscope slides, X-rays, MRI scans, and other medical examination processes, which are some of the most promising scientific fields. In light of this, a study [13] talks about a revolution in the field of comprehensive spine surgery care, which is still good and improving and can reduce surgeon stress, improve technical accuracy, and assist in predictive analytics for surgeons. It also helps improve patient selection before surgery.

With the massive amounts of data made possible by medical imaging, researchers can teach computers to recognize patterns associated with conditions like pneumonia, cancer, or a wrist fracture that is difficult for a person to see. This is done by transferring the data to "artificial neural networks".

The system then operates based on an algorithm or a set of rules; the interpretation becomes easier and more accurate with more data. Numerous medical applications currently use this technique, which is called deep learning. A study [14] confirmed that eye surgery requires sensory and motor skills that approach human limits, and yet machine learning-enhanced surgical robots could ultimately improve retina surgery and potentially change the system.

In a related medical use, Google has previously developed tools to help ophthalmologists identify eye conditions in diabetic people, and pathologists diagnose cancer by analyzing microscope slides. According to a study [15], people with prostate cancer are receiving care more quickly, thanks to the growing use of machine learning for diagnosis and treatment planning.

According to studies, screening lowers the chance of dying from lung cancer. Aside from identifying specific tumours, screening can also discover areas potentially developing into cancer. This allows radiologists to classify patients into risk exposure groups and ascertain if they require skin tissue tests or ongoing monitoring of areas likely to be affected.

But radiographic testing has shortcomings, according to researchers, as it can miss tumours or misidentify benign areas as malignant ones, sending patients to dangerous procedures like lung tissue testing or surgery. Different radiologists may view this matter differently depending on how they examine the scan itself.

"The entire experiment process is like a student in school," stated Dr. Tsai. Until students can comprehend what cancer is and what will or will not be cancer in the future, we use a sizable quantity of data for training, teaching, and pop quizzes. After spending a lot of time teaching, we gave them a final exam on data we had never seen before, and the outcome we observed on the final exam was good" [16].

A study [17] concluded that surgical practice would change completely in the future with the use of artificial intelligence and robots in diagnostic, radiology, pathology, and ultimately, supervision of independent surgery without allowing the decline of surgical skills of the surgeon's hand and emphasizes the need for surgeons to be prepared to adopt more intelligent training methods.

Many people, including healthcare professionals, fear that robots and artificial intelligence systems will replace their occupations despite the remarkable technological advances and the emergence of increasingly intelligent technologies. According to existing signs, integrating artificial and human intelligence is the best approach to bring about the impending transformation in the healthcare industry [18].

Therefore, a study concludes that clinically meaningful surgical robots are likely to be achieved by the end of the 21st century and that combining artificial intelligence allows surgical robots to improve outcomes to achieve optimal care. According to a technology and future-focused website, surgical robots, particularly the Da Vinci

system, extend the surgeon's control over the device from a remote control unit inside the operating room. A study noted that the Da Vinci system developed an operating and recording system that can benefit surgeons when evaluating surgical skills and teaching trainees [19]. This can provide the best results by combining artificial intelligence with clinical examination.

One of the most ambitious procedures, and one that is said to be the first in the world, took place in Montreal in 2010, where the first joint performance between a robotic surgeon and a robotic anesthesiologist (dubbed "McSleepy" as a kind of pun) was achieved, and the data collected on this procedure reflected the excellent performance of these robotic doctors.

To evaluate the safety of robotic surgery, the Massachusetts Institute of Technology looked back at data from the FDA more than ten years after surgical robots were first used in operating rooms. 1,391 patient injuries and 144 patient deaths were reported during the study period, the majority of which were brought on by equipment malfunctions or technical issues.

The summary stated, "despite the relatively high number of reports, the vast majority of procedures were successful and did not involve any problems." Nonetheless, compared to specialities like gynaecology and general surgery, the frequency of occurrences in the most complex surgical fields (such as cardiovascular surgery) was "much higher".

It seems this situation is similar to fast food; while robotic surgery can be performed well in some specialities, human surgeons should perform complex surgical procedures, at least for now. However, this situation can quickly change, and robotic surgeons may become more capable of working independently of human surgeons. Then, it will become difficult to determine who will be held liable when errors occur.

In this context, a study focused on legal aspects and classified responsibility into the issue, responsibility, and blame, where the last element (blame) was the least clear, as it cannot be imagined in the current state of technology. Moreover, many patients may not agree to be ready to relinquish control of their lives through surgical robots. Therefore, can a patient sue a robot for malpractice?Even though the technology in these circumstances may still be relatively young, the litigation is legally ambiguous. Experts typically believe that medical malpractice results from a doctor's negligence or a breach of a particular standard of care. Still, there are unanswered questions. Can the patient's relatives hold the human surgeon in charge of the robot accountable?

Will the manufacturer of the robot face the consequences? Or will the engineer who created the robot be held accountable? Although there are no satisfactory solutions for these problems, they must be addressed as soon as possible.

## 5 Previous Studies

Most studies have emphasized the importance of collaboration between doctors, engineers, and technicians in improving, developing, and supporting artificial intelligence and robotic surgery without replacing or lowering the level of human surgeons.

Robotic surgery offers high precision and skill, low error rates, more accurate diagnoses, better results, less risk of blood loss, faster wound healing, and faster recovery. Therefore, the researcher concluded from previous studies that surgical robots represent a significant upcoming revolution that requires everyone (doctors and patients) to accept, deal with, and help spread and develop them.

Note that responsibility and the patients' ability to accept surgical robots were classified. Studies [1, 12], on the other hand, have shown that increasing artificial intelligence capabilities and surgical robots' ability to predict and diagnose treatment can improve outcomes and increase access to desired healthcare. Studies [9, 14] have also discussed how the use of big data and its development in radiology, imaging, and data analysis can benefit surgeons in evaluating their technical skills contributing to surgical training and decision-making.

A study [10] considered the Internet of Things (IoT) and healthcare systems as global data support collected from medical devices. Most studies used a descriptive-analytical approach to provide an overview of the artificial intelligence and surgical robot revolution in healthcare in previous,

current, and future years. They also pointed out some of the difficulties facing this technology and encouraged cooperation from everyone to benefit from it. In a study [11] in this paper, it was noted that responsibility and the patients' ability to accept surgical robots were classified.

Studies [1, 12], on the other hand, have shown that increasing artificial intelligence capabilities and surgical robots' ability to predict and diagnose treatment can improve outcomes and increase access to desired healthcare. Studies [9, 14] have also discussed how the use of big data and its development in radiology, imaging, and data analysis can benefit surgeons in evaluating their technical skills contributing to surgical training and decision-making.

A study considered the Internet of Things (IoT) and healthcare systems as global data sources collected from medical devices. Most studies used a descriptive-analytical approach to provide an overview of the artificial intelligence and surgical robot revolution in healthcare in previous, current, and future years. They also pointed out some difficulties facing this technology and encouraged everyone's cooperation to benefit from it. We will now review previous studies from the perspective of their strengths and weaknesses and our perspective.

A study focused on the legal, regulatory, and ethical frameworks for developing future artificial intelligence and surgical robot standards. The discussion has pushed forward the potential for independent surgical robotics and AI, focusing on ethics, regulations, and legal aspects (such as civil law, liability, international law, privacy, product/device legislation, and malpractice).

Responsibility was classified into several legal aspects as follows: (1) the issue, (2) responsibility, and (3) blame. These three aspects were addressed when discussing responsibility for artificial intelligence and independent surgical robots. We believe that shortly, surgical robots may learn and perform routine surgical tasks that a human surgeon can later supervise.

If robots acquire human citizenship, these responsibilities could need to be reviewed. Blame is the least evident outcome of this, as it is unimaginable given the state of technology now. Furthermore, many patients might not be prepared to give up control over their lives and security to steel arms, fingers, and electronic circuits. Employment is also in jeopardy. Professional associations may oppose autonomous surgical robotics and AI applications.

A paper on artificial intelligence and robots: A Mix Transforms the Operating Room summarised the existing evidence in the literature regarding the AI techniques used during robotic surgery. According to the study, the application of AI in robotic surgery is anticipated to improve the surgical experience throughout the procedure and have a major influence on future surgical training.

Both seek to improve the quality of surgical care through precise surgery, which calls for multidisciplinary collaboration between software developers, engineers, and surgeons. According to a recent age study, prostate cancer detection, prediction, diagnosis, and treatment have become faster with the growing application of AI and machine learning.

The effective training, operation, and regulation of artificial intelligence-based decision support applications necessitates cooperation between urologists, data scientists, computer researchers, and engineers. Doctors cannot be replaced by artificial intelligence; nevertheless, physicians who acquire and apply artificial intelligence skills will take the position of conventional physicians who fail to adapt to the rapidly emerging digital landscape.

A study on artificial intelligence and robotic eye surgery questioned whether we are ready for this technology. The author mentioned that precise surgery of the specific retina involves sensory and motor skills that approach the limits of human physiology for stability, accuracy, and the ability to detect small forces.

Despite assumptions favouring robots in surgery and development efforts, machine learning-enhanced robotic systems may improve retina surgery with robot assistance and potentially change the system. Many obstacles prevent end users from adopting robots, including cost, size, functional limitations, accuracy, human acceptance, and, most importantly, superior safety outcomes. A study [12] discusses artificial intelligence and the future of surgical robotics.

The author describes clinical trials of artificial intelligence in the surgical context, providing insight into the surgical future. Clinically

meaningful surgical robots will likely be developed by the end of the 21st century, combining artificial intelligence with human expertise to increase surgical capability, improve outcomes, and increase access to desired care.

Independent surgical devices must be highly developed; however, they must also be relatively developed and provide technical or financial advantages over traditional surgical technologies to be widely used. A study [9] discusses the digital surgeon and how big data, automation, and artificial intelligence will fundamentally change surgical practice in the future. The surge of big data and artificial intelligence will radically change surgical practice in the future.

The study emphasizes the importance of using artificial intelligence and robots in diagnosis, radiology, and pathology while ultimately supervising independent surgery without allowing the degradation of manual surgical skills. The study recommends that Surgeons must also be prepared to adopt more intelligent training methods, as the possibility of losing manual surgical skills is high.

Robots and artificial intelligence in dentistry were the subject of a study [4]. Data-driven algorithmic analysis has been used in dentistry to improve diagnosis, treatment planning, and dental decision-making by analyzing vast volumes of data using artificial intelligence and machine learning. This review seeks to offer fresh perspectives to the community and advocates for a greater application of these cutting-edge contemporary technologies, sometimes known as dentronics.

Only a few robotic applications,mostly restricted to experimental use cases,have been implemented thus far. A study [13] discusses the potential of artificial intelligence (AI) and robots in comprehensive spinal surgery. Most published literature on AI in spinal surgery focuses on predictive analytics and radiographic imaging. AI has enormous potential to revolutionize comprehensive spinal surgery care.

In research, AI computation can collect, process, and analyze patient data to extract valuable clinical information for studies. While still new and improving, robotic-assisted surgerycanreduce surgeon fatigue and improve technical precision. Predictive surgical analytics can help improve patient selection before surgery, surgical indicators, and individualized care after surgery. Spinal surgery is constantly evolving.

AI and machine learning in the surgical domains are the subject of a study [2]. AI and machine learning have been used to evaluate surgical technical proficiency. These methods can follow the surgeon's eye and body movements, identify patterns in video records, detect machine movement, and assess the surgeon's cognitive abilities.

A mechanism for recording and replay has been created for the conventional da Vinci surgical equipment. Surgeons can assess trainees' surgical abilities and impart knowledge using this technology. Combining AI and physician clinical examination can lead to better results and more efficient shared decision-making. Telemedicine is another use of AI and machine learning that has helped provide care to underserved communities.

Machine learning requires large data sets to program computers to create algorithms. There is also the possibility of misclassifying data points that do not follow the typical patterns learned by the device. Further studies are needed to determine feasibility, effectiveness, and cost. The introduction of robots in surgical settings during COVID-19 is the subject of a study [17].

The extent to which human interaction is necessary for our activities has been made clear by the current pandemic. This requirement presents a major obstacle in COVID-19. During periods of heavy patient flow, the employment of robotics and AI can help reduce infectious contamination and help patients manage surgical surroundings.

Since robotic surgery can be technically performed in an operating room with just the patient present, it would also be the best situation for preventing the spread of germs. This transformed how surgery was performed, and hospitals are now under unprecedented pressure to address the present pandemic.

A study [15] on Artificial Intelligence in Digestive System Diseases: In this study, researchers addressed several questions, such as: where are we headed? Gastroenterology is a field where artificial intelligence can have a significant impact. With the help of artificial intelligence, image analysis can provide more accurate evaluation and information than traditional analysis.

Artificial intelligence and robots can also assist surgeons in performing digestive system operations. Technological advancements are rapidly expanding in digestive system diseases and surgery. Researchers must continue to work on new artificial intelligence technology and human-machine interfaces to improve diagnosis and warning accuracy.

A study [18] concerning the use of AI in surgery: Technological developments in imaging, navigation, and automation are bringing about a progressive transformation in surgical practice through artificial intelligence. Artificial intelligence capable of learning complex tasks independently with minimal initial training data will be crucial for next-generation systems.

Robots that follow a gradual path represented by different levels of autonomy are evolving, and most specialists will turn to diagnosis and decision-making. We still have a long way to go to match robotic surgery with the level of surgical intelligence. Clearly, different issues must be addressed before artificial intelligence is smoothly integrated into the future of surgery.

The main focus of a study [14] reviewing medical artificial intelligence was using AI in machine learning, robotics, image recognition, and expert systems, among other medical domains. It also covered issues and developments in various past and present fields. Due to globalization, many research institutes worldwide have conducted many studies on this subject in recent years.

As a result, artificial intelligence in medicine has advanced significantly and has a bright future. By now, the number of publications on artificial intelligence has grown quickly, especially in China, which has steadily emerged as a pioneer in the area thanks to high-speed network transmission. Stable, dependable, and secure remote collaborative surgery can be guaranteed with remote technical guidance [20].

Surgical hazards can be efficiently reduced by experts who understand the patient's condition and the procedure's progress. Future developments in artificial intelligence are anticipated to provide more difficulties, especially in data mining and machine learning. The review of artificial intelligence for the Internet of Things and medical systems, which encompasses the application and practice of artificial intelligence methodology in many healthcare domains, was covered in a study [10] on artificial intelligence, the Internet of Things, and enhanced medical systems.

The review demonstrates that artificial intelligence techniques are applied in medicine in four primary areas: robotic surgery, personalized therapy, predictive approaches, and heart disease diagnosis. These techniques are mostly applied to patient data analysis for better health outcomes.

Robotic surgery technologies, such as automated endoscopic systems and robotic oral surgery, have significant benefits because they reduce patient risk, yield better blood loss outcomes, and hasten recovery. Personal healthcare systems and the Internet of Things (IoT) can support medical device data globally. The use of blockchain for IoT in medicine is a broad field.

## 6 Conclusion and Recommendations

Through reviewing previous studies, we can conclude that the support provided by artificial intelligence technologies played a pivotal role for doctors, particularly in supporting more accurate disease diagnoses and reducing the psychological pressure and physical effort required to reach their goals.

They were designed to help doctors rather than replace their essential role, resulting in positive impacts on strengthening communication and interaction between doctors and patients, raising patient morale, improving surgical accuracy and reducing surgical errors, accelerating recovery, and providing good levels of healthcare without replacing the skills of human surgeons.

Training surgeons to use machines, teaching them how to input medical data, and combining the efforts of technical and medical experts will open up a vast world of information, which will lead to the development of intelligent applications in areas such as remote diagnosis and surgery disease prediction. In the near future, it is possible that a surgeon could be a surgical robot that determines your appointment based on your condition and is the only one with you in the operating room.

Artificial intelligence and robots constitute the future of the healthcare sector in the Kingdom of Saudi Arabia, according to Vision 2030. We are

moving in a good direction so far. Still, it is necessary to conduct and disseminate studies on artificial intelligence and surgical robots within academic circles, especially among doctors, technicians, and the media, to promote the concept of artificial intelligence and the healthcare services it will provide, which everyone desires.

Therefore, we will conclude this paper with some recommendations:

1. Developing independent surgical devices and providing them with technological and financial advantages to revolutionize robotic surgeries.
2. Machine learning training should be increased through long-term repetition to match robotic surgery methods with those of surgeons in surgical procedures.
3. Raising patient awareness about surgical robots and their numerous benefits.
4. Teaching surgeons about using artificial intelligence and surgical robots from the early years of medical studies.
5. Organizing competitive competitions at universities, especially in technical, engineering, and medical departments, on artificial intelligence and surgical robots to better promote surgical robot culture in scientific circles.
6. Establishing huge medical information platforms that doctors and surgeons can benefit from in diagnosis and treatment planning.
7. Private and charitable institutions collaborate with researchers and developers to accelerate this technological revolution of artificial intelligence and surgical robots and contribute to the country's healthcare development.

Using surgical robotic technologies, the article probably explains how AI may improve surgical operations, leading to better accuracy, efficiency, and patient outcomes. It may look into how AI algorithms might help with diagnosis, evaluate medical data, and even do certain activities independently with some guidance from healthcare providers. To further use AI in healthcare, especially surgical robots, the article will likely conclude with recommendations for more research, funding, and implementation strategies.

These recommendations may address the need for multidisciplinary cooperation between healthcare providers and AI specialists, as well as ethical concerns and regulatory obstacles. By reviewing these recent studies from 2011-2023, we found that it is essential to develop and collaborate with doctors and experts in artificial intelligence and patients to improve healthcare level, quality, and efficiency.

We are on the cusp of a great and massive revolution in artificial intelligence and surgical robots, which is moving at a relatively rapid pace in various healthcare fields. This necessitates our acceptance and dissemination of it and accelerating its education and training so that we can reach the ambitious level of Vision 2030 in healthcare. We must not neglect the obstacles this technology will face, but by collaborating with specialists, we can facilitate those obstacles.

## References

1. **Algethami, R. F., Alotbi, M. F., Alsulaimani, A. I., Alkhaldi, L. M., Alhossaini, Z. A., Alzahrani, K. M. (2023).** Assessment of knowledge and awareness regarding robotics-assisted surgery (RAS) among saudi population: a cross sectional study. Medical Science, Vol. 27, No. 132, pp. 1–9. DOI: 10.54905/disssi/v27i132/e73ms2853.
2. **Sultan, I., Bardi, M. F., Baatta, A. M., Almaghrabi, S., Mohammed, R. A. (2022).** Medical students' attitude towards robotic surgery: A cross-sectional survey. Journal of Medical Education and Curricular Development, Vol. 9, pp. 238212052110664. DOI: 10.1177/23821205211066483.
3. **Talib, H. S., Almurashi, A. A., Alzahrani, K. T. (2022).** Cardiopulmonary resuscitation CPR quality outcome of patients with cardiac arrest by using robotics/artificial intelligence in hospitals. International Journal of Innovative Research in Medical Science, Vol. 8, No. 09, pp. 440–444. DOI: 10.23958/ijirms/vol07-i09/1502.
4. **Alzahrani, A. A. H. (2024).** Perceptions and attitudes of dental practitioners toward robotic dentistry and artificial intelligence in saudi arabia. AIP Advances, Vol. 14, No. 4. DOI: 10.1063/5.0204058.

5. **Moglia, A., Georgiou, K., Georgiou, E., Satava, R. M., Cuschieri, A. (2021).** A systematic review on artificial intelligence in robot-assisted surgery. International Journal of Surgery, Vol. 95, pp. 106151. DOI: 10.1016/j.ijsu.2021.106151.
6. **Andras, I., Mazzone, E., van-Leeuwen, F. W. B., De-Naeyer, G., van-Oosterom, M. N., Beato, S., Buckle, T., O'Sullivan, S., van-Leeuwen, P. J., Beulens, A., Crisan, N., D'Hondt, F., Schatteman, P., van-Der-Poel, H., Dell'Oglio, P., Mottrie, A. (2019).** Artificial intelligence and robotics: A combination that is changing the operating room. World Journal of Urology, Vol. 38, No. 10, pp. 2359–2366. DOI: 10.1007/s00345-019-03037-6.
7. **Park, J. J., Tiefenbach, J., Demetriades, A. K. (2022).** The role of artificial intelligence in surgical simulation. Frontiers in Medical Technology, Vol. 4, pp. 1076755. DOI: 10.3389/fmedt.2022.1076755.
8. **Panesar, S., Cagle, Y., Chander, D., Morey, J., Fernandez-Miranda, J., Kliot, M. (2019).** Artificial intelligence and the future of surgical robotics. Annals of Surgery, Vol. 270, No. 2, pp. 223–226. DOI: 10.1097/sla.0000000000003262.
9. **Yousef, D. A. A., AlKhazali, Z., Qawasmeh, R. A., Alshamayleh, H. Z. (2023).** Medical invention marketing strategies on buying: Surgical medical robot. International Journal of Data and Network Science, Vol. 7, No. 2, pp. 647–656. DOI: 10.5267/j.ijdns.2023.3.007.
10. **Oniani, S., Marques, G., Barnovi, S., Pires, I. M., Bhoi, A. K. (2020).** Artificial intelligence for internet of things and enhanced medical systems. Bio-inspired Neurocomputing, Studies in Computational Intelligence, Vol. 903, pp. 43–59. DOI: 10.1007/978-981-15-5495-7_3.
11. **Ergin, E., Karaarslan, D., Şahan, S., Bingöl, Ü. (2023).** Can artificial intelligence and robotic nurses replace operating room nurses? The quasi-experimental research. Journal of Robotic Surgery, Vol. 17, No. 4, pp. 1847–1855. DOI: 10.1007/s11701-023-01592-0.
12. **Barkati, N., Ntefeh, N., Okasha, A., Takshe, A. A., ElKhatib, R., Chelli, S. (2023).** Robotic assisted surgery in the united Arab emirates: Healthcare experts' perceptions. Journal of Robotic Surgery, Vol. 17, No. 6, pp. 2799–2806. DOI: 10.1007/s11701-023-01716-6.
13. **Gangadharan, K., Nesa-Kumari, G. R., Dhanasekaran, D., Malathi, K. (2020).** Detection and classification of various pest attacks and infection on plants using recursive backpropagation neural network with GA based particle swarm optimization algorithm. Indonesian Journal of Electrical Engineering and Computer Science, Vol. 20, No. 3, pp. 1278. DOI: 10.11591/ijeecs.v20.i3.pp1278-1288.
14. **Liu, R., Rong, Y., Peng, Z. (2020).** A review of medical artificial intelligence. Global Health Journal, Vol. 4, No. 2, pp. 42–45. DOI: 10.1016/j.glohj.2020.04.002.
15. **Sung, J. J., Poon, N. C. (2020).** Artificial intelligence in gastroenterology: Where are we heading? Frontiers of Medicine, Vol. 14, No. 4, pp. 511–517. DOI: 10.1007/s11684-02-07424.
16. **Urias, M. G., Patel, N., He, C., Ebrahimi, A., Kim, J. W., Iordachita, I., Gehlbach, P. L. (2019).** Artificial intelligence, robotics and eye surgery: Are we overfitted? International Journal of Retina and Vitreous, Vol. 5, No. 52, pp. 1–4. DOI: 10.1186/s40942-019-0202-y.
17. **Zemmar, A., Lozano, A. M., Nelson, B. J. (2020).** The rise of robots in surgical environments during covid-19. Nature Machine Intelligence, Vol. 2, No. 10, pp. 566–572. DOI: 10.1038/s42256-020-00238-2.
18. **Zhou, X., Guo, Y., Shen, M., Yang, G. (2020).** Application of artificial intelligence in surgery. Frontiers of Medicine, Vol. 14, No. 4, pp. 417–430. DOI: 10.1007/s11684-020-0770-0.
19. **Mansingh, P., Pattanayak, B. K., Pati, B. (2022).** Big medical image analysis: Alzheimer's disease classification using convolutional autoencoder. Computación y Sistemas, Vol. 26, No. 4, pp. 1491–1501. DOI: 10.13053/cys-26-4-4090.
20. **Mathur, R. P., Sharma, M. (2023).** A multi-objective task scheduling scheme GMOPSO-BFO in mobile cloud computing. Computación y Sistemas, Vol. 27, No. 2, pp. 477–488. DOI: 10.13053/cys-27-2-3953.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Mohammed bin Qassim Al-Asi.*

ISSN 2007-9737

# Web-based Application Layer Distributed Denial-of-Service Attacks: A Data-driven Machine Learning Strategy

K Alluraiah*, Manna Sheela Rani Chetty

Koneru Lakshmaiah Education Foundation,
Department of Computer Science & Engineering,
India

allura02@gmail.com, sheelarani_cse@kluniversity.in

**Abstract.** DDoS attacks, which aim to overwhelm a system with requests, are commonplace in the cyber world. In this type of assault, bandwidth and processing resources are deliberately clogged to disrupt legitimate users' interactions. These attacks inundate the victim's system with packets, rendering it inaccessible. Diverging from the singular source of Denial of Service (DoS) attacks, DDoS attacks emanate from many servers, magnifying their impact. Over the last decade, a concentrated effort has been invested in comprehending the orchestration and authentication of DDoS attacks, resulting in valuable insights into discerning attack patterns and suspicious activities. Currently, the focus has shifted towards real-time detection within the stream of network transactions, constituting a critical research domain. Yet, this focus often sidelines the importance of benchmarking DDoS attack assertions within the streaming data framework. As a remedy, the Anomaly-based Real-Time Prevention (ARTP) framework has been formulated and designed specifically to combat application layer DDoS attacks, particularly targeting web applications. Employing advanced machine learning techniques, ARTP offers adaptable methodologies to swiftly and accurately pinpoint application-layer DDoS attacks. Rigorous testing on a representative LLDoS (Low-Level DoS) benchmark dataset has affirmed the resilience and efficiency of the proposed ARTP model, underscoring its capacity to achieve the research objectives set forth.

**Keywords.** Detection of App-DDoS, denial of service (DoS) attacks, application layer DDoS (App-DDoS), LLDoS dataset, distributed DoS (DDoS) attacks.

## 1 Introduction

Without question, in the present setting, the Internet has established itself as a crucial component in the realm of daily business activity. The extraordinary evolution of communication methods, trade practices, and individual online presence continues to captivate attention [1]. With this dynamic setting, it's no surprise that Internet vulnerabilities like Denial of Service (DoS) attacks and their more sophisticated sibling, Distributed Denial of Service (DDoS) attacks, have become major problems.

A DoS attack targets a specific system, often a web server, to disrupt its capacity to serve legitimate users without affecting the network or server architecture. DDoS assaults use a network of devices spread over the Internet to flood targeted systems with excessive traffic, draining their resources and disrupting network integrity, making the systems unavailable to legitimate users.

Network layer attacks that utilize DDoS and application layer HTTP attacks are the primary forms of distributed loss of service attacks. Both forms are common in the realm of decentralized denial of service attacks. One type of attack involves an adversary utilizing methods such as IP spoofing to flood the system in question with fake data packets [2]. Application layer DDoS assaults, also known as App-DDoS attacks, flood the system being attacked with numerous legitimate requests to disrupt its functioning.

The attacker's infected machines must create legitimate TCP links with the victim's system to avoid the connections being terminated [3]. One instance is the Internet flood, which uses increased HTTP requests to overwhelm the targeted servers. Operating at the program level with the ground, these assaults target the hosting server of websites by overwhelming it with a large number of

simultaneous and ostensibly valid requests, ultimately making the service inaccessible to genuine users.

Recent studies, like [4], explore new methods to evade DDoSdefences and reveal methodologies that enable more effective assault partnerships. The HTTP flood method increases the number of HTTP requests to attack weaknesses in the targeted servers' design. During this flood of data, the servers cannot distinguish between genuine payload information and the overwhelming amount of seemingly normal HTTP requests. The main contribution of the paper is:

- Desigining the Anomaly-based Real-Time Prevention (ARTP) framework for prediction DDoS attacks in targeting web applications.
- Employing advanced machine learning techniques, ARTP offers adaptable methodologies to swiftly and accurately pinpoint application-layer DDoS attacks.
- Experimental results have been implemended and the suggested ARTP model increases efficiency, accuracy and prediction ratio compared to existing models.

## 2 Literature Review

Certain DDoS protection methods designed to combat HTTP floods rely on application layer knowledge, as a recent survey shows [5]. One example is the DDoS shield [6], which uses the detection of session beginning timings and control of time delays between arrivals to prevent HTTP floods.

The study "Using Page Access Leads to Counter the HTTP Flood" explores the complex connection between data sizes and surfing durations. However, these methods are ineffective when dealing with the high packet transmission rates enabled by the organized actions of malevolent Botnets [7].

Implementing a CAPTCHA-based probability validation method may be beneficial in some instances, but it might overly complicate the user experience [8]. This might lead to an unintentional Denial of Service situation, especially when it is crucial to answer practical complaints [9].

The Concealed semi-Markov modelling (HSMM) is presented in reference [10] to use the characteristics of transactional demand instances to decipher common client access patterns. This model assesses the continuous increase of customers utilizing HSMM-derived data. This method results in many false warnings due to the variety in how users access the system, particularly given the different ways individuals browse.

Clients interacting with external internet interfaces, entering URLs for searches, or using various browsers may unintentionally cause false alerts, introducing confusion. The main goal of the suggested technique is to evaluate transactional relationships using regular and flood data. Unlike traditional benchmarking methods, our suggested model is based on thorough observations of the request stream over a long period.

Eric Muhati and DandaRawat [11] suggested the Data-Driven Network Anomaly Detection with Cyber Attack and Defense Visualization. The author employs a data analytics-based intrusion detection system that examines all network connections; this innovative method is based on network scanning tools and network discovery services, which enable us to view the network according to the number of active IP-based networking devices.

The author then takes precautions by graphically differentiating between safe and dangerous connections, using red for the former and blue for the latter. The experimental assessment reveals that the model outperforms previous research in this field with an impressive F1 score of 97.9% and a low false positive rate of 0.3%.

Talha Farid and MaheyzahSirat [12] recommended the Hybrid of Supervised Learning and Optimization Algorithm for Optimal Detection of IoT Distributed Denial of Service Attacks. An authenticated and verified real-time IoT traffic dataset is used to train machine learning classification algorithms, and their prediction performance is first evaluated in this work.

With a prediction accuracy of 97%, the findings show that Logistic Regression (LR) is the most effective supervised machine learning classifier for identifying IoT DDoS assaults. As a result, Grasshopper Optimizer Algorithms (GOA) emerge

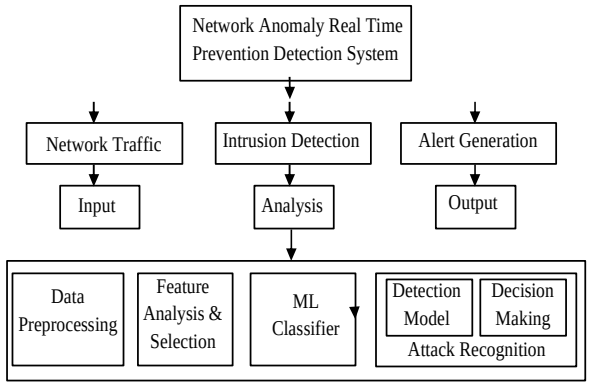



**Fig. 1.** The architecture of network anomaly real time prevention detection system

as the top optimizer for enhancing LR's prediction accuracy to 99% in a subsequent study on LR hybridization with optimization algorithms.

Therefore, the best option for detecting IoT DDoS attacks is the LR hybridized by GOA. Therefore, the paper lays forth A data-driven strategy to counter new forms of malicious IoT DDoS attacks, such zero-day attacks.

Wubetu Barud Demilie and Fitsum Gizachew Deriba [13] discussed the structured query language injection (SQLI) attack detection using machine learning and hybrid techniques.

The Keras library was used to implement the ML algorithms, while the Tensor FlowLearn package was used to implement the classical techniques.

We used 54,306 bits of data from weblogs, cookies, session use, and HTTP (S) request flows to train and test our model for this suggested research study.

In terms of performance assessment findings for training sets, measures like f1-score (99.35% and 99.57%), accuracy (99.20% and 99.60%), recall (99.65% and 99.61%), and hybrid technique (ANN and SVM) outperform other ML approaches.

The next parts of this work carefully adhere to a structured path. Section 3 explains the ARTP model, and Section 4 thoroughly analyzes a test study to assess the effectiveness of ARTP. Section 5 concludes this academic pursuit by providing definitive insights.

## 3 Web-based Applications and Regional Denial-attacks: Utilizing a Data-driven Learning-based Approach

We have developed an anomaly-based Actual Time Preventive (ARTP) framework to address the threat of Application Layer DDoS assaults on the World Wide Web. The multitude of demands prompted the creation of this framework. The system's prompt and early identification of application-level DDoS assaults has been a major success, establishing it as one of its most notable accomplishments.

The ARTP model's core resides in its multifarious entity metrics, including important aspects like request chain length, packet count variations, packet sorting variants, request intervals, and the contextual background of transactional chains. Many common benchmarking approaches rely on sessions or requests as main inputs; however, they unintentionally overlook the essential premise of this methodology.

Comprehensive testing was conducted using the LLDoS dataset to assess the usefulness and strength of the proposed model, highlighting its flexibility and resilience. The anticipated approach demonstrates exceptional effectiveness in dealing with application layer DDoS assaults. With the current environment of online applications, the volume of received requests has increased to the size of petabytes, far beyond the gigabyte-level load of prior web request systems.

Using labelled datasets, various supervised learning algoithms, including Neural Networks, Support Vector Machines (SVMs), and Random Forests, are learned to distinguish between valid and dangerous inflow requests.

These algorithms can accurately identify and classify recognized DDoS attacks because they learn torecognize patterns indicative of these attacks using characteristics taken from the traffic data. This study uses supervised and unsupervised learning methodsto spot suspicious behaviours in network data, which might indicate new types of distributeddenial of serviceattacks.

To be more precise, thisstudy finds clusters of requeststhat are much out of the ordinary by using

**Table 1.** Specifics on the normal and DDoS attack time frames, and the training and testing datasets used

| Number of SuccessfulTransactions (CS) | | | | 2,29,386 |
|---|---|---|---|---|
| | | Training (60%) | Testing (40%) | Total |
| N (Normal) $CS_N$ | Theoperations | 74,260 | 49,520 | 123,780 |
| | Sessions | 2872 | 1838 | 4710 |
| | Groups | 287 | 175 | 462 |
| | tf: thelength of the time frame | 608 | 614 | 1222 |
| | Many time frames | 242 | 147 | 389 |
| D (DDoSAttack) $CS_D$ | Theoperations | 56,440 | 35,961 | 92,401 |
| | Sessions | 2548 | 1533 | 4081 |
| | Groups | 253 | 154 | 407 |
| | tf: thelength of the time frame | 744 | 759 | 1503 |
| | Many time frames | 264 | 167 | 431 |

a clustering-based anomaly detection method. As part of ourun supervised learning strategy, this study clusters incoming requests according to variables, including request frequency, pay load characteristics, and IP addresses.

Our approach can adapt to new threats and identify attack patterns not present in the training data since this study used unsupervised learning methods. Because of this, ourDDoS detection techniqueis more resilient and can with stand new attacks at the web applicationlayer.

## 4 Determining the Duration of a Period

The abbreviation CS represents a collection of Client Sessions denoted as {s1, s2,…,sn}. Every session in this set, which is labelled as {llsi^siCS}, includes transactions classified as either N (typical) or D (disorganized, indicating a DDoS assault) [14].

The aggregate count of transactions in CS encompasses both regular (CSN) and anomalous (CSD) transactions, forming the composite volume of transactions associated with DDoS attacks.

The heuristic measures outlined in Section 3 will undergo validation through experimentation by employing these datasets. To clarify further, the dataset CS, which encompasses both CSN and CSD instances, undergoes a crucial partitioning process into distinct CSN (normal) and CSD (DDoS attack) subsets.

Subsequently, harnessing the K-Means technique [15], the sessions within these subsets are individually clustered. This strategic clustering aids in the determination of the optimal number of clusters within the CSN and CSD sets. This culminates in the simultaneous computation of time frames for both CSN and CSD, adding a layer of precision to the analysis process.

Selecting characteristics to detect and respond to DDoS attacks effectively includes the web applications' session behaviour, request patterns, and pay load content . Features that are important for distributed denial of service attacks are given priority. These features include request mazda mx5rates at peak hours, pay load sizes larger than typical, and sudden increases in session creation.

These aspects may capture the unique patterns of traffic from DDoS attacks. In Figure 1, the web-based application layer works on a network Anomaly real-time Prevention Detection System; here, the ARTP system can be categorized into three network system models.

The first model will be designed to detect the DDoS attack on the network traffic models that can be represented as input models, and the second intrusion detection system model will be represented as the network analysis model [16].

The third network model will be represented as a DDoS attack detection generation alert message; this means whenever the attack is detected from the network application layer, an alert message will be generated and sent to the ARTP system, So this is the output model. In the analysis model, feature

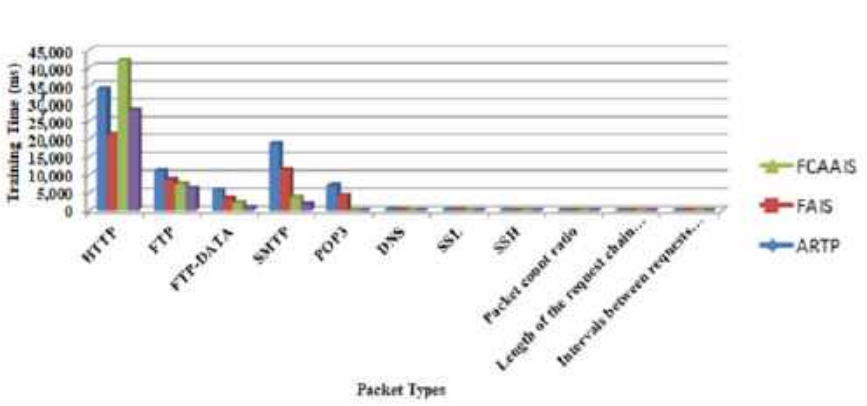



**Fig. 2.** Comparative analysis of the request for CSN tests and CSD training as a proposition

selections include data pre-processing, feature analysis and selection, machine learning classifier and attack recognition, i.e., DDoS attack detection model and decision making.

The above network models will be given the detection of DDoS attack rate and reduced attack rate by machine learning classifiers.

This paper outlines the procedures used to pre-process test data to train machine learning models to identify distributed denial-of-service (DDoS) attacks at the web-based application layer. Data formatting, normalization, and optimization for analysis are all aspects of pre-processing that guarantee accurate results.

Within the context of the amalgamated regular transaction group $CS_N$, let $C=\{c_1, c_2,\ldots,c_m\}$ denote clusters characterized by a spectrum of K values. Within the realm of each cluster $C_j\{c_j \exists c_j \in C\text{^ } j= 1, 2, 3,\ldots M\}$, the determination of time frames entails a calculation rooted in the discrepancy between the completion time of the most extensive session and the initiation time of the most succinct session.

Considering the cluster $C_j$, let $SB_N(C_j) = \{sb_1, sb_2, \ldots sb_{|cj|}\}$ represent an array of session start times arranged in ascending order. Here, sb1 corresponds to the session with the earliest start time, while $sb_{|cj|}$ corresponds to the one with the latest initiation time. Additionally, denote $SE_N(c_i)= \{se_1, se_2,\ldots se_{|ci|}\}$ as a sequence encompassing session end times within cluster $c_i$, with se1 marking the termination of the shortest session and $se_{|ci|}$ denoting the end of the longest one. The ensuing expression encapsulates the time frame of cluster $c_j$ ($tf(c_j)$):

$$tf(c_j)=\sqrt{(se_1 - se_2)^2}. \quad (1)$$

The standard length of a time frame is computed as follows, taking into account all clusters:

$$<tf(c_j)>=\frac{\sum_{j=1}^{M} tf(cj)}{M}. \quad (2)$$

The Absolute Deviation (tfAD) of time frames for all clusters is defined as:

$$tfAD=\frac{\sqrt{\sum_{j=1}^{M} (<tf(C)>-tf(Cj))2}}{M}. \quad (3)$$

Finally, the time frame (tf) is determined by multiplying the average length of the time frame (<tf (C) >) with the Absolute Deviation of the time frame (tfAD):

$$tf=<tf(C)>+tfAD. \quad (4)$$

## 5 The Heuristics of Empirical Metrics

### 5.1 Request Chain Length (RCL)

In the context of a DDoS attack, the assailants orchestrate numerous queries directed at the Web server. Conversely, a tactic involving substantial HTTP requests aims at compromising websites; furthermore, such attacks can arise when genuine users inadvertently submit requests that vastly surpass typical sizes.

To preempt possible memory-related challenges on the server end, the range of requests within specific time frames is delineated

**Table 2.** The metric's values, as well as packettype ratios for different packet types

| Types of packets | The ratio between packet types | | | | | |
|---|---|---|---|---|---|---|
| | N (normal) $CS_N$ | | | D (DDoS Attack) $CS_D$ | | |
| | Training | | Test | Training | | Testing |
| | Minimum limit | Maximum limit | | Minimum limit | Maximum limit | |
| HTTP | 0.4384 | 0.46 | 0.36 | 0.68 | 0.75 | 0.77 |
| FTP | 0.1348 | 0.16 | 0.15 | 0.12 | 0.13 | 0.12 |
| FTP-DATA | 0.0627 | 0.07 | 0.07 | 0.02 | 0.05 | 0.02 |
| SMTP | 0.2368 | 0.28 | 0.23 | 0.053 | 0.07 | 0.07 |
| POP3 | 0.0982 | 0.11 | 0.08 | 0.0 | 0.0 | 0.0 |
| DNS | 0.0025 | 0.004 | 0.03 | 0.0 | 0.0 | 0.0 |
| SSL | 0.0009 | 0.003 | 0.03 | 0.03 | 0.05 | 0.47 |

**Table 3.** Metrics for ARTP performance and actual out comes

| | | | | | |
|---|---|---|---|---|---|
| True positive (tp) | The "normal" transactions are routine ones. | | | | 36,531 |
| False Positive (fp) | The total number of legitimate transactions that were incorrectly marked as intrusions. | | | | 4244 |
| True Negative (tn) | How many spoofed transactions are spoofed. | | | | 45,144 |
| False Negative (fn) | The percentage of unapproved transactions that are mistakenly classified as regular. | | | | 541 |
| Precision | $\frac{tp}{tp+fp}$ | 0.897 | Accuracy | $\frac{(tp+tn)}{(tp+tn+fp+fn)}$ | 0.944 |
| Recall/sensitivity | $\frac{tp}{tp+fn}$ | 0.985 | F-Measure | 2× (precision * recall)/ precision + recall | 0.938 |
| Specificity | $\frac{tn}{fp+tn}$ | 0.914 | | | |

for both N (normal) and D (DDoS attack) scenarios, culminating in the creation of CSN and CSD.

Within CSN, $TS(CS_N) = \{ts_1, ts_2, \ldots ts_{|TS(CSN)|}\}$ dissects the progression of transactions within the normal set into distinct time frames. Here, $ts_j$ represents the number of transactions received during the $j^{th}$ time frame, while tf denotes the temporal extent of the time frame, as elucidated in Section 3.1. The following model is utilized to calculate the average chain length of transactions observed across all time frames within CSN:

$$<TS(CS_N)> = \sum_{j=1}^{|TS(CSN)|}\{|ts_j| \exists\ ts_j \in TS(CSn)\}. \quad (5)$$

Within every CSN time frame cluster, the Absolute Deviation (clAD) of request chain length is computed using the following formula:

$$clAD = \frac{\sqrt{\sum_{j=1}^{|TS(CSn)|}\left(<TS(CSn)> - (|tsj| \exists tsj \in TS(CSn))\right)2}}{|TS(CSn)|}. \quad (6)$$

Within each time frame of the CS collection, the Request Chain Length (RCL) is determined by amalgamating the average transaction chain <TS(CSN)> with the Absolute Deviation of RCL (clAD):

$$rcl(CS_N) = <TS(CS_N)> + clAD. \quad (7)$$

### 5.2 Packet Count Ratio

DDoS attacks deploy substantial bandwidth volumes to inundate their targets, rendering them inaccessible.Packet computing is crucial in data flow for tasks such as detecting counterfeit attacks,

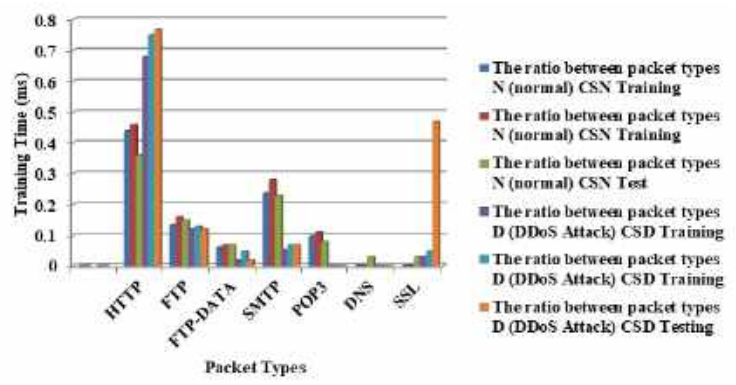



**Fig. 3.** Different packet types of metric values and packet type ratios

identifying unspoofedDDoS attacks, and functioning as a meter for flow coherence.

This calculating method assesses the packet processing level in the order of requests in normal and DDoS attack scenarios by analyzing the message volume during each period.

Determining if a DDoS assault has occurred relies on analyzing the volume of packets for each time increment.

The computation is governed by the number of packets per time frame, represented as tsj, which belongs to the set TS(CSN) and ranges from 1 to the total number of elements in TS(CSN). This reveals the complex relationship between particles and query patterns:

$$\mathrm{rp}(\mathrm{tj_i}) = \sum_{j=1}^{|TS(CSn)|} \frac{|P(tsj)|}{\sum_{j=1}^{|TS(CSn)|} |P(tsk)|}. \quad (8)$$

The period Floor Packets Support Relative Difference (tflpsAD) duration is calculated for every frame in the Argument Space Net (CSN) using the approach outlined below:

$$\mathrm{tflpsAD} = \frac{\sqrt{\sum_{j=1}^{|TS(CSn)|} (1 - rp(tsj))2}}{|TS(CSn)|}. \quad (9)$$

The total amount of packets in a CSN reflects the network's activities throughout all periods, while the Time Frame Tier Messages Support Relative Deviation offers a thorough assessment of particle intensity.

The time frame for Levels Packets Supported Absolute Deviation is obtained by adding these numbers:

$$\mathrm{rpc}(\mathrm{TS}(\mathrm{CS_N})) = \frac{\sum_{j=1}^{|TS(CSn)|} rp(tsi)}{|TS(CSn)|} + tflpsAD. \quad (10)$$

### 5.3 Intervals between Request Ratios

The time elapsed between consecutive requests in order within a single session is calculated as the access time, starting from creating training-focused transactional sets. Each session produces a series of time frames for assessing the time intervals between requirements in regular situations and during DDoS assaults. Within CSN, every frame includes an Interval Total Deviation (iAD), calculated as follows:

$$\mathrm{iAD} = \frac{\sqrt{\sum_{j=1}^{|TS(CSn)|} (gm(CSn) - lm(tsj))2}}{|TS(CSn)|}. \quad (11)$$

In conclusion, the degree of interval is ascertained by aggregating the mean value computed across all CSN intervals, along with including the Interval Absolute Deviation (iAD) concerning intervals within the training set CSN. This holistic approach encapsulates a comprehensive evaluation of interval dynamics:

$$\mathrm{ri}(\mathrm{CS_N}) = \frac{\sum_{j=1}^{|TS(CSn)|} lm(tsj)}{|TS(CSn)|} + \mathrm{iAD}. \quad (12)$$

### 5.4 Packet Type Ratio in a Predetermined Time Frame

The term "packet ratio threshold" refers to the fraction of unique packet types, such as DNS,

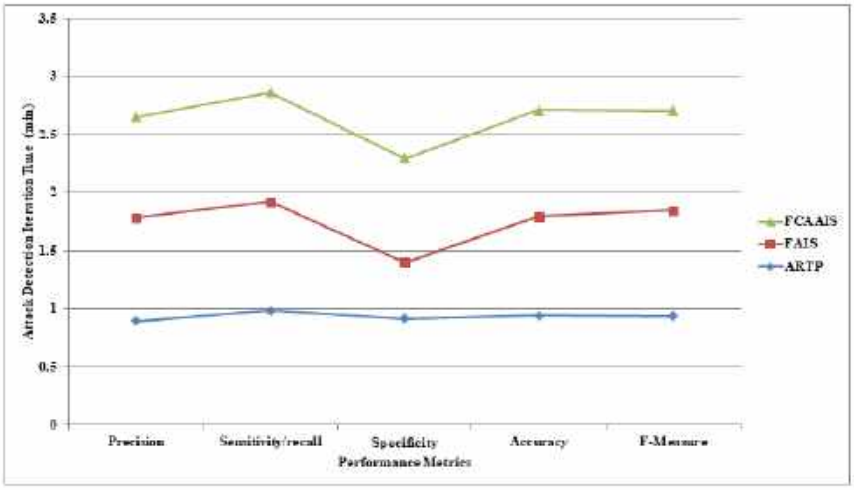



**Fig. 4.** A comparison is presented between FCAAIS, FAIS and ARTP

SMTP, FTP-DATA, SSL, POP3, HTTP, and FTP, that occur within a given period that is less than the fraction of unique packet types, such as FTP-DATA, POP3, HTTP, DNS, SMTP, and FTP, that occur within the transactions of N (normal) as CSN and D (DDoS attack) as CSD.

For each packet type included in the order of requests analyzed within $ts_l$, the local support within CSN, represented by $ts_i$ {$ts_i \exists\ ts_i \in TS(CS_N)$ ^ i= 1,2,3,…|$TS(CS_N)$|} of $CS_N$, establishes the threshold. Moving on, for every type of packet, represented by $pt_k$ {$pt_k \exists\ pt_k \in PT$^ k= 1,2,…,|PT|}, the Absolute Deviation (ptsAD) of packet type support is investigated across all time frames within $CS_N$, juxtaposing local and global support. This evaluation unfolds in the following manner:

$$\text{ptsAD(ptk)} = \frac{\sqrt{\sum_{i=1}^{|TS(CSn)|} (gs(ptk) - lstsi(ptk))2}}{|TS(CSn)|}. \quad (13)$$

In the concluding phase, the degree of ptk (rpt(ptk))is determined through the summation of the average packet threshold type and the Absolute Deviation of packet type support (ptsAD). This computation follows the sequence of time lengths within the training set CSN, encompassing various packet types:

$$\text{rpt}(pt_k) = \frac{\sum_{i=1}^{|TS(CSn)|} \text{lstsi(ptk)}}{|TS(CSn)|} + \text{ptsAD}(pt_k). \quad (14)$$

### 5.5 Order of Requests or Request Chain Context

Given the ensemble of generated transactions designated for training, the progression involves segmenting the requests into discrete time frames corresponding to regular and DDoS attack instances. a request pair set $rps_N$ is formulated within the training transaction set CSN, encompassing pairs $rps_N = \{p_1, p_2, \ldots, p_{|rpsN|}\}$, where each pair $p_i$ represents the immediate two consecutive requests within the CSN sequence.

The local support $ls_{tsj}$ (pi) attributed to $ls_{tsj}$ ($p_i$) of $p_i$ ($p_i \exists\ p_i \in rps_N$ ^ i=1, 2,…, $|rps_N|$) quantifies the occurrences of the pair pi within time frame $ts_j$.

The training dataset $CS_N$ includes all recorded sequences of requests, and for each pair $p_i$ ($p_i \exists\ p_i \in rps_N$ ^ i=1,2,…,$|rps_N|$), the cumulative support $gs(p_i)$ includes all instances of the pair.

This iterative process culminates in a comprehensive understanding of request pair dynamics. When do these conditions shift from supporting pi locally to supporting it globally?

The Absolute Deviation (rpsAD), an abbreviation of relative standard deviation, is calculated for each pi ($p_i \exists\ p_i \in rps_N$ ^ i=1, 2,…,$|rps_N|$), This calculation is expressed as follows:

**Table 4.** FAIS and FCAAIS are use DTO compare the suggested ARTP technique

| | ARTP | FAIS | FCAAIS |
|---|---|---|---|
| Precision | 0.938 | 0.911 | 0.855 |
| Sensitivity/recall | 0.944 | 0.851 | 0.917 |
| Specificity | 0.915 | 0.496 | 0.885 |
| Accuracy | 0.985 | 0.935 | 0.942 |
| F-Measure | 0.895 | 0.889 | 0.869 |

$$\text{rpsAD}=\sqrt{\frac{\sum_{j=1}^{|TS(CSn)|}\left(gs(pi)-lstsj(pi)\right)2}{|TS(CSn)|}}. \quad (15)$$

The mean value derived from all the pair carry values is computed across the sequence of time frames within the training set $CS_N$. Moreover, for every possible pairing that arises within the scope of this paper, the Absolute Deviation (rpsAD) of request pair support is explored. The sequence ($p_i \exists\ p_i \in rps_N \wedge i=1,2,\ldots,|rps_N|$) is being evaluated in parallel on the request chain context $rcc(p_i)$:

$$rcc(p_i)=\frac{\sum_{j=1}^{|TS(CSn)|} lstsj(pi)}{|TS(CSn)|}+\text{rpsAD}. \quad (16)$$

Organizations can enhance their ability to detect and mitigate web-based application layer DDoS attacks, minimize disruption to their online services, and protect themselves from potential security threats by utilizing attack recognition techniques, deploying detection models based on machine learning, and building adaptive decision-making processes.

## 6 Outcomes of Experiments and Performance Evaluation

The tests have come to a close to evaluating the suggested model ARTP's, resilience, process complexity, Scalability and detection accuracy.

### 6.1 Experimental Results

To simulate application layer DDoS attack scenarios, the framework LLDOS 2.0.2 [17] is harnessed under normal and attack conditions. For the testing phase, 229,386 transactions are processed, encompassing N (normal) and D (disruptive) transactions representative of DDoS attacks.

60% of this dataset has been set aside for training, while the remaining 40% will be used for testing.Our study uses synthetic transactional data imitating web application traffic as test data. A custom-built data-generating tool was used to produce these transactions, which were meant to simulate user interaction swith web-based applications.

Assessed independently using the CS dataset, each metric is generated, comprising CSN (normal) and CSD (DDoS attack) instances. CSN comprises a total of 123,780 transactions, out of which 60 percent (74,260) are dedicated to training, and the remaining 40 percent (49,520) are reserved for testing.

Sessions in the CSN [18] training and testing datasets are evenly divided into 1-minute and 10-second intervals. As a next step, the K-Means technique is applied separately to both the training and testing phases to establish the locations of any existing clusters.

The proposed K means technique's efficiency is compared with other unsupervised learning techniques, such as Principle Component Analysis and the Apriori algorithm.

Similarly, the assault dataset CSD is divided into sessions and clusters using the same manner to facilitate training and testing. Additionally, temporal frames (as detailed in Sect. 3.1) are synthesized from the stream of sessions, subsequently defining the temporal scope of the CSN and CSD training and evaluation phases in Table 1.

### 6.2 Request Chain Length (RCL)

The request chain length is characterized by the maximum count of requests originating from a transaction. This concept is expanded in the CSN and CSD training environment to incorporate the typical length of requests recorded from clients within a time frame described as either an attack or a normal scenario.

The goal is to better equip pupils to deal with real-world difficulties. A comprehensive overview of these details is meticulously presented in Table 2.

### 6.2.1 Packet Count Ratio

Within the training set, the ratio of packet counts is calculated by tallying the total number of packets received during each interval along the request chain, which is split into CSN and CSD. A chain of requests undergoes this analysis. All the while the CSN and CSD teams are being trained and tested, a thorough examination of the packet calculating process is being conducted.

FTP-DATA, SSL, HTTP, POP3, DNS, and SMTP packets are only a few that fall inside this inquiry's scope. These computations encompass calculating the packet ratio across all time frames within CSN and CSD. The detailed results of these computations are meticulously presented and recorded in Table 2.

### 6.2.2 The Ratio of Request Intervals

The table 2 summarises extracted metrics from the training sets in depth. Request Chain Length (RCL), Request Interval Ratio, Packet Count, and Packet Count Ratio are all metrics that may be measured across CSN and CSD.

These metrics have been extracted from the designated training datasets and are presented in Tables 2(a) & (b). "approach time" is used in CSN and CSD transactions to describe the time between the first and last requests made within a given session. For each period, the ratio of intervals between requests is calculated by painstakingly calculating the approach time for each pair of requests within the same session.

CSN and CSD packet types, request chain length (RCL), packet count ratio, and RCL are all measured using the provided training sets. In Figure 2, the representations of types of packets were tested on normal attack of CSN packet count of N and DDoS attack of CSD packet count D. The resultant graph will be analyzed on training and testing of CSN packet count of N and CSD packet count of D.

### 6.2.3 Ratio of Packets Types

Training and testing procedures that use the CSN and CSD datasets allow for examining ratios associated with threshold creation for each packet type. Internet Protocol (HTTP), File Transfer Protocol (FTP) (including FTP-DATA), Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), and Secure Sockets Layer (SSL) packets are all included here.

The required ratios are detailed in Table 2 below. These tables owe a great deal to the CSN and CSD training sets. This set of ratios encapsulates dissimilar packet types constituting a substantial portion of commonly employed packet types within the application layer.

## 6.3 Performance Analysis

Precision, sensitivity (true positive rate), specificity (true negative rate), accuracy, and F-measure are some of the key parameters that can be used to assess the usefulness of ARTP's detection capabilities. Quantitative parameters can include accuracy, sensitivity (the proportion of correct diagnoses), and specificity (the proportion of incorrect diagnoses).

Similarly, a quantitative parameter is specificity. As showcased in Table 4, this procedure's statistics offer insights into its performance. Predicted requests are accurately classified as normal when they are, and expected attacks are correctly labelled as attacks.

As determined in the experiments, sensitivity guides the identification of true positives projected as positive outcomes. Meanwhile, true negatives that are accurately anticipated as negatives are attributed to specificity. Accuracy is also notable, signifying the duration required for precise request classification.

In other words, the evaluation shows a lower risk of misidentifying a normal condition as an attack than misclassifying an attack configuration as normal; hence the sensitivity rating is higher than the specificity rating. The evaluation shows that the sensitivity is greater than the specificity.

An attack misclassified as a normal request has a 1-sensitivity rate of 0.0145, whereas misclassifying a normal request as an attack has a 1-specificity rate of 0.0859. There has been a significant performance boost for each of these numbers.

Consequently, it is reasonable to assert that the suggested ARTP demonstrates increased effectiveness in detecting attacks, as evident from the numerical performance metrics and practical observations in Table 3. In contrast, the Fuzzy Artificial Immune System FAIS [19] and feature

correlation analysis and association impact scale (FCAAIS) [20-21] models are proposed for DDoS attack detection.

Conducted on the same dataset, these models exhibit scalability and resilience in forecasting the scope of network DDoS attacks, achieving an approximate detection accuracy of 91%. However, when juxtaposed with the proposed ARTP model, these approaches encounter challenges stemming from process complexity, particularly regarding the statistical metrics employed for performance calculation. Notably, as displayed in Table 4 and Fig. 4, the precision of our proposed ARTP model surpasses both FAIS and FCAAIS, exemplifying higher predictive accuracy.

In Figure 3, the resultant graph will depict different packet types of metric values and packet type ratios. All packets compared to the ratios between the packet types of normal attack CSN and DDoS attack CSD with minimum and maximum limits will be done in the testing phase. Finally, the resultant graph will showthe accuracy of all types of packets.

Figure 4 compares the ARTP with FCAAIS and FAIS based on precision, recall, specificity, F-measure and accuracy. These metrics, including precision, recall, specificity, F-measure, and accuracy, were employed to evaluate the efficacy of each model in various classification or pattern recognition tasks [22, 23].

## 7 Conclusion

This work suggests an approach to machine learning based on empirical data to identify, protect against, and stop denial of provider (DDoS) assaults at the app level in actual time on various web services. The author of this essay diligently attempts to classify the article's contributions into three separate categories.

The initial contribution is to evaluate particular metrics of a demand stream to determine if the intent is malicious or benign. Experimental threshold values derived from the measurements in the ARTP framework are used to determine if a stream's behaviour meets the criteria for being classified as a flood.

The second contribution focuses on observing ARTP behaviour, supported by thorough testing on the LLDoS dataset. This component combines perceived procedural complexities with appropriate speed to enhance detection accuracy. The results show that the indications derived from the training data set, which is used to assess the status of the request stream, are very promising and essential.

The final addition uses threshold values from the training set Report Phrase to identify if a request stream could indicate a software layer DDoS assault over time. ARTP's resilience is thoroughly tested under simple settings and high speeds. The technique outlined in this research shows a surprising capacity to maintain high forecast accuracy while also reducing computing workload, proving its efficiency in operation.

## References

1. **Mahdavi-Hezavehi, S., Rahmani, R. (2020).** An anomaly-based framework for mitigating effects of DDoS attacks using a third party auditor in cloud computing environments. Cluster Computing, Vol. 23, No. 4, pp. 2609–2627. DOI: 10.1007/s10586-019-03031-y.
2. **Krishna-Kishore, P., Ramamoorthy, S., Rajavarman, V. (2023).** ARTP: Anomaly based real time prevention of distributed denial of service attacks on the web using machine learning approach. International Journal of Intelligent Networks, Vol. 4, pp. 38–45. DOI: 10.1016/j.ijin.2022.12.001.
3. **Byers, S., Rubin, A. D., Kormann, D. (2004).** Defending against an internet-based attack on the physical world. ACM Transactions on Internet Technology, Vol. 4, No. 3, pp. 239–254. DOI: 10.1145/1013202.1013203.
4. **Estevez-Tapiador, J., Garcia-Teodoro, P., Diaz-Verdejo, J. (2005).** Detection of web-based attacks through Markovian protocol parsing. 10th IEEE symposium on computers and communications, Vol. 2396, pp. 457–462. DOI: 10.1109/iscc.2005.51.
5. **Kishore, P. K., Rajavarman, R. V. (2023).** Detection of DDoS enabled flood attacks using an ensemble classifier in distributed networks. International Journal of Intelligent Systems

and Applications in Engineering, Vol. 11, No. 3, pp. 578–590.

6. **Yatagai, T., Isohara, T., Sasase, I. (2007).** Detection of HTTP-GET flood attack based on analysis of page access behavior. IEEE Pacific rim conference on communications, computers and signal processing, pp. 232–235. DOI: 10.1109/pacrim.2007.4313218.
7. **Sivatha-Sindhu, S. S., Geetha, S., Kannan, A. (2012).** Decision tree based light weight intrusion detection using a wrapper approach. Expert Systems with Applications, Vol. 39, No. 1, pp. 129–141. DOI: 10.1016/j.eswa.2011.06.013.
8. **Shevtekar, A., Ansari, N. (2009).** Is it congestion or a DDoS attack? IEEE Communications Letters, Vol. 13, No. 7, pp. 546–548. DOI: 10.1109/lcomm.2009.090628.
9. **Fouladi, R. F., Ermiş, O., Anarim, E. (2022).** A DDoS attack detection and countermeasure scheme based on DWT and auto-encoder neural network for SDN. Computer Networks, Vol. 214, pp. 109140. DOI: 10.1016/j.comnet.2022.109140.
10. **Kishore, P. K., Ramamoorthy, D. S., Rajavarman, D. (2019).** Detection, defensive and mitigation of DDoS attacks through machine learning techniques: A literature. International Journal of Recent Technology and Engineering, Vol. 8, No. 4, pp. 2719–2725. DOI: 10.35940/ijrte.d7335.118419.
11. **Muhati, E., Rawat, D. (2024).** Data-driven network anomaly detection with cyber attack and defense visualization. Journal of Cybersecurity and Privacy, Vol. 4, No. 2, pp. 241–263. DOI: 10.3390/jcp4020012.
12. **Farid, T., Sirat, M. (2023).** Hybrid of supervised learning and optimization algorithm for optimal detection of IoT distributed denial of service attacks. International Journal of Innovative Computing, Vol. 13, No. 1, pp. 1–12. DOI: 10.11113/ijic.v13n1.329.
13. **Demilie, W. B., Deriba, F. G. (2022).** Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques. Journal of Big Data, Vol. 9, No. 1. DOI: 10.1186/s40537-022-00678-0.
14. **Wisanwanichthan, T., Thammawichai, M. (2021).** A double-layered hybrid approach for network intrusion detection system using combined Naive Bayes and SVM. IEEE Access, Vol. 9, pp. 138432–138450. DOI: 10.1109/access.2021.3118573.
15. **Xie, Y., Shun-Zheng, Y. (2009).** A large-scale hidden semi-Markov model for anomaly detection on user browsing behaviors. IEEE/ACM Transactions on Networking, Vol. 17, No. 1, pp. 54–65. DOI: 10.1109/tnet.2008.923716.
16. **MIT (1998).** 1998 DARPA intrusion detection evaluation dataset. Darpa Intrusion Detection Evaluation. https://www.ll.mit.edu/ideval/data/1998data.html.
17. **Hartigan, J. A., Wong, M. A. (1979).** Algorithm AS 136: a K-means clustering algorithm. Applied Statistics, Vol. 28, No. 1, pp. 100. DOI: 10.2307/2346830.
18. **Powers, D. M. (2006).** Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. 23rd international conference on machine learning, pp. 37—63. DOI: 10.48550/arXiv.2010.16061.
19. **Jyothsna, V., Rama-Prasad, V. (2016).** FCAAIS: Anomaly based network intrusion detection through feature correlation analysis and association impact scale. ICT Express, Vol. 2, No. 3, pp. 103–116. DOI: 10.1016/j.icte.2016.08.003.
20. **SenthamilSelvan, R., Wahidabanu, R. S. D., Karthik, B. (2020).** Intersection collision avoidance in dedicated short-range communication using vehicle ad hoc network. Concurrency and Computation: Practice and Experience, Vol. 34, No. 13. DOI: 10.1002/cpe.5856.
21. **Gelbukh, A., Pérez-Alvarez, D. A., Kolesnikova, O., Chanona-Hernandez, L., Sidorov, G. (2024).** Multi-instrument based N-grams for composer classification task. Computación y Sistemas, Vol. 28, No. 1. DOI: 10.13053/cys-28-1-4903.
22. **Adebanji, O. O., Ojo, O. E., Calvo, H., Gelbukh, I., Sidorov, G. (2024).** Adaptation of transformer-based models for depression

detection. Computación y Sistemas, Vol. 28, No. 1. DOI: 10.13053/cys-28-1-4691.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is K Alluraiah.*

ISSN 2007-9737

# Multi-label Classification of IoTData Stream: A Survey

Mashail Althabiti*, Manal Abdullah, Omaima Almatrafi

King Abdulaziz University,
Faculty of Computing and Information Technology,
Saudia Arabia

malthabiti0001@stu.kau.edu.sa, {maaabdullah, olamatrafi}@kau.edu.sa

**Abstract.** The overall number of Internet of Things (IoT) devices is rapidly growing, generating a massive amount of continuous data stream. The data stream is arriving at a rapid speed, potentially unbounded, which has emerged due to smart services and advanced technologies. Data stream classification is a challenging task that must fulfil stream constraints such as limited memory, a single scan of data, and real-time response. In many emerging applications, stream instances could be associated with more than one class label, as when predicting a given movie genre, different labels may be given: action, horror, adventure, or all, and this refers to Multi-label Classification (MLC). This review mainly aims to review the literature on the multi-label classification task from 2014 to 2023. It examines state-of-the-art versatile MLC methods in general data streams and methods utilized for IoT applications, which are considered one of the main sources of data streams generated by IoT devices. It also focuses on two main challenges: class imbalance and concept drift. It encapsulates the well-known MLC tools and datasets utilized for this task. Moreover, it highlights the gaps that need further attention in future research.

**Keywords.** Multi-label classification, concept drift, class imbalance.

## 1 Introduction

A collection of linked smart objects that can self-configure as a dynamic network is known as the Internet of Things (IoT) [1]. It allows the interaction between humans and objects anytime, anywhere, generating massive data streams. The data stream refers to a series of data instances characterized by rapid speed, huge volume, drifting nature, and non-stop arrival [2].

Such data involve valuable information and patterns crucial for many applications and can be discovered using Machine Learning (ML) methods. ML is an Artificial Intelligence (AI) based method that can learn from the experience without being explicitly programmed [3].

The application of ML methods on IoT data stream is found in many applications in different domains such as medicine, economy, entertainment, education, smart cities, and many others. For example, smart cities utilize IoT and ML methods to address urban issues and meet citizens' needs. It has been used in smart homes, waste management, tourism, and smart transportation to facilitate daily tasks or improve productivity.

One ML approach that fits in numerous IoT and smart city tasks is classification, especially in monitoring and control applications [4]. Classification is a supervised ML method that learns from old labelled data and predicts the label of the incoming data stream instances, assuming that only a single label is produced [5].

However, there are numerous real-life situations and applications where data must be classified into multi-label. For example, the text categorization task is considered the primary motivator for Multi-Label Classification (MLC) [6]. Documents can be assigned to several subjects or categories.

MLC may also be used in many application domains, such as emotion recognition, medical diagnosis, image annotation, music/movie categorization, etc. MLC has attracted much research attention due to its significance in several applications of different domains.

However, the MLC of the data stream has some challenges, including limited memory, a single scan of data, and concept drift. Concept drift is the

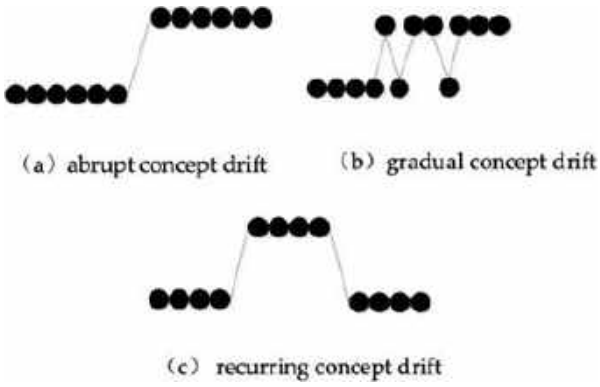



**Fig. 1.** Concept drift types based on speed [6]

underlying data stream distribution change leading to poor predictions [7]. Such changes can occur abruptly, gradually, or recurrent over time. MLC methods must be able to diagnose when performance degrades and repair to recover to a stable status. In addition, MLC data could arrive with skewed class proportions, another challenge that deteriorates the classifier performance called class imbalance.

Among the different MLC challenges, this paper focuses on class imbalance and concept drift since they are the main characteristics of the data stream. It provides a review of multi-label classification methods used in data streams, generally and in IoT, to identify the state-of-the-art methods in the literature and highlight the gaps to be explored in future work. It mainly answers the following questions,

- Q1: What versatile multi-label classification methods are used to process data streams?
- Q2: What multi-label classification methods were proposed to deal with IoT?
- Q3: Do multi-label classification methods in the literature address Concept Drift and Class Imbalance?
- Q4: What are the main multi-label classification datasets and tools?

The remainder of this article is structured as follows. Section 2 provides a brief background about related concepts.

Section 3 provides the methodology used to conduct the review. Section 4 provides the results of the literature questions. Finally, Section 5 presents the discussion and conclusion.

## 2 Background

The data stream is a series of instances, $DS = (x_1, y_1), (x_2, y_2), (x_3, y_3) \ldots (x_i, y_i)$ that arrive continuously, where x is the input vector representing a set of features, and y is the corresponding label.

In other words, the data stream is a massive, non-stop sequence of instances arriving at high speed. Nowadays, data streams are generated at a higher speed than ever before due to technological advances. Its high speed, chronological order, infinite and immense volume, and dynamical changes characterize it.

An example of the data stream is NASA satellites generating 4 Terabytes (1012 Bytes) of streaming images in only one day . Also, millions of query streams are processed daily by Google and Yahoo, two popular search engines. The demand for versatile data services is expected to increase by 2050, as it is estimated that around seven billion people will live in urban centres and smart cities.

Smart Cities are where citizens can live in an urban setup surrounded by an intelligent network of interconnected devices and information data that become part of their daily routine.

The connectivity between the main places and services is enabled, such as transportation networks, utility and public services, health care systems, and others, improving citizens' quality of life. Proliferating technologies like Internet of Things (IoT) sensors, cloud computing, machine learning, smartphones, and security systems work hand in hand to build smart cities.

The Internet of Things (IoT) is a network of physical objects or devices integrated with sensors and other technologies. This enables real-time data streams between these devices to be exchanged over the Internet.

By 2030, it is anticipated that 50 billion Internet of Things (IoT) sensors and gadgets will be connected to the Internet over high-capacity, low-latency networks.

As a result, online learning algorithms and intelligent systems require the Internet of Things.

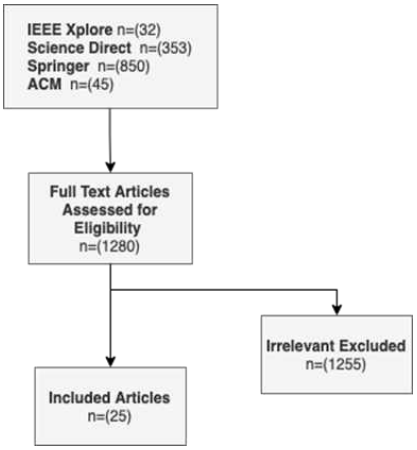



**Fig. 2.** Steps to select the articles in the literature

Multi-label classification (MLC) is one of the challenges of the data stream. In single-label classification, each data instance is assumed to be assigned to one class. On the contrary, MLC associates a data instance with a set of labels. The multi-label classifier can be defined as $F: x \rightarrow Y$.

The primary function of $F$ is to take the input as any instance $x \in X$, where $X$ is the input space and produce a set of labels $Y \in L$, where $L$ is a set of disjoint labels [8]. An example of an MLC application is image classification, a prominent data type in smart cities collected by IoT devices [9].

Multi-classification algorithms can automatically annotate the input images with appropriate keywords. That would help categorize the streaming images and use them in appropriate applications. The following subsections will present an overview of two main challenges of MLC inthe data stream.

### 2.1 Concept Drift

Concept drift is one of the data stream phenomena representing the changes in the statistical properties of the features or class labels over time [6].

Formally, the concept drift occurs when the Pt $(x, y) \neq P\_(t + \Delta)\ (x, y)$, where $P(x, y)$ is the joint distribution between data attributes and labels at time t. Concept drift is characterized by the speed in three types, as illustrated in Figure 1, detailed below:

- Abrupt drift Occurs when the underlying data distribution changes suddenly at a particular time with a new concept; therefore, the model built before the drift is unreliable.
- Gradual drift: Occurs when data is alternated between two concepts, meaning new data takes time to substitute the old one.
- Recurrent drift: Occurs when an old concept reappears once or multiple times in the future.

An example showing the importance of detecting concept drift is using a classification model with a drift detector in the Industrial Internet of Things (IIoT). In IIoT, many gadgets are connected [10].

All the machine conditions are collected and shown in a sensor administration system so that the production administrator can ideally control all machines' operations in realtime.

However, machine components may fail or age over time, so the drift detector analyzes the real-time data and alerts when such an abnormal situation occurs. The administrator can replace the damaged components quickly and avoid defective manufacturing products.

### 2.2 Class Imbalance

Class imbalance is another phenomenon existing in the MLC of the data stream. It refers to the dataset with skewed class proportions, so the classes of large proportion are called the majority classes [11,12]. In contrast, the classes with a smaller proportion are the minority classes.

Due to the presence of majority classes, the classifier tends to ignore minority classes; therefore, the classifier's performance degrades. Class imbalance occurs in several real-world scenarios, such as real-time network monitoring systems, where the classification model must learn from the data streams with skewed class distributions [23].

## 3 Research Methodology

To conduct the review, five steps are carried out: (1) Define the scope, (2) Search of the articles, (3) Select the articles, then (4) analyze the articles, and finally (5) present the results of the review

[14]. In the first step, the scope of the review is identified. It involves the definition of articles' inclusion and exclusion criteria.

Also, the databases to be searched are selected, and the search terms are formulated. Inclusion and exclusion criteria are mainly defined to set the boundaries of the review and select only articles that fulfil the criteria.

In this research, the inclusion criteria are: (1) selected articles should be written in English, (2) explicitly study the MLC, and (3) the date of publishing should be between 2014 and 2023. Thus, to select the articles included in the review, each should satisfy these criteria; otherwise, it will be excluded. Generally, machine learning could span many disciplines, such as engineering, neuroscience, and mathematics.

This study reviews the literature on multi-label classification from a computer science point of view. Also, four databases have been selected to search for the articles: IEEE Xplore, ScienceDirect, ACM, and SpringerLink. The impact factor of ScienceDirect (Elsevier) Lancet is 202.731, Association for Computing Machinery (ACM) is 16.6, SpringerLink is 2.6. The second step involves searching for the scientific articles in the identified databases using the search terms.

The search terms used to look for the article in selected databases are Multi-label Classification, Concept Drift, Class Imbalance, and Internet of Things (IoT). AND and OR operators have been used to link some terms, which are: (Multi-label classification AND Concept drift), (Multi-label classification AND Class Imbalance), and multi-label classification AND IoT OR Internet of Things). Articles should include one of these search terms to be selected.

In the third phase, the articles are filtered according to the criteria. Only relevant articles are selected, which is done by reading the abstract, title and sometimes the full text.

Also, the citations of each selected article are checked to enrich the sample. Figure 2 shows the steps to select the articles to be included in the literature.

In the fourth step, the selected articles are analyzed, and the findings and insights related to the literature questions and scope are highlighted. Finally, the fifth and last step in conducting the review is to present the analysis results, showing the findings and insights.

## 4 Results

Twenty-five articles have been selected to be included in the literature. Figure 3 shows the number of publications per year. The results are divided into three subsections as follows:

### 4.1 Q1: What Versatile Multi-label Classification Methodscan Process Data Streams?

MLC algorithms in a non-stationary environment can be classified into three approaches: Problem Transformation, Algorithm Adaptation and Ensemble [5].

Problem Transformation (PT) simplifies and transforms the MLC problem into simpler single-label classification problems. Thanh et al. [15] believe incremental learning applies to the massive data stream.

They developed a MLC method for drifting streams based on the Bayesian method. The method considers the correlation between the labels to predict the set of labels. It also uses the Hoeffding inequality to set the number of predicted labels. It employs a decay function to detect the drifts, giving recent data more weight.

The method has been assessed in a stream environment and a stationary environment. Braytee et al. [16] have proposed an integrated multi-label classification method called (ML-CIB).

ML-CIB mainly tackles the label incompleteness and class imbalance. It utilizes the label regularizer to handle the class imbalance and learns a new label matrix to handle the missing label.

The authors have also introduced a multi-label feature selection for an effective, relevant feature selection. Ding et al. [17] proposed an algorithm for processing multi-label imbalanced data based on majority and minority assessment. It eliminates the majority of classes whose number exceeds the penalty value as a preprocessing method.

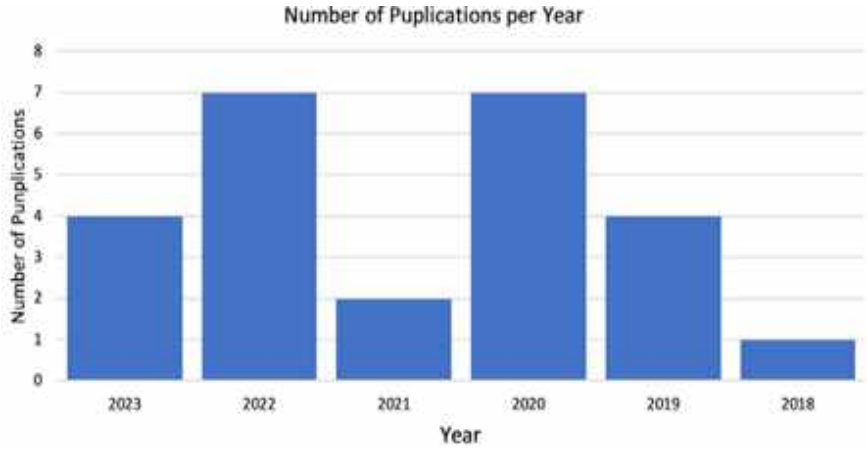



**Fig.3.**The number of publications per year

It also employs a metric controlling the cost of the majority class and the value of the minority class during the classification process.

The proposed algorithm uses a decision function generated by the binary relevance and chain classifier to classify multi-label imbalanced data. Ji et al. [18] proposed a framework called (ML-INC) addressing different issues in multi-label learning, including class imbalance, incomplete labels, irrelevant features, and noisy data.

To this end, ML-INC leverages low-rank and sparse decomposition and high-order label correlation, consistency, and regularization. Algorithm Adaptation (AA) is another MLC approach extending traditional classification algorithms to cope with the multi-label issue.

Roseberry et al. [19] presented a punitive k nearest neighbours algorithm with a self-adjusting memory (MLSAMPkNN) designed to deal with multi-label data streams with drift. MLSAMPkNN employs majority vote kNN on a self-adjusting sliding window containing the current concept.

It adapts the changes within the data stream efficiently using a punitive system that removes the errant data instances early from the window. The proposed algorithm has been evaluated, proving its versatility for diverse learning scenarios.

Like MLSAMPkNN, Roseberry et al. [20] have attracted attention to the issue of drifting data streams in MLC. They highlighted the need to predict the complete set of labels for multi-label instances. The authors offered a self-adjusting k nearest neighbours algorithm (MLSAkNN) to cope with these tasks and solve the mentioned issue. In contrast to previous algorithms used for such streams and instances, their algorithm uses a collectively working variety of mechanisms to deal with the issue of multi-label data.

Their algorithm works with the value k, which is self-adapting for each label and can successfully deal with multi-label instances. To address the problem of imbalanced labels, Rastogi and Mortaza [21] have suggested a multi-label classifier called Linear Imbalance Multi-label data learning with Label Specific Features (IMLSF).

IMLSF uses label correlation to increase accuracy and the structural information of the data through data locality to guarantee that similar data instances have similar class labels. To overcome class imbalance, IMLSF additionally uses a weighting mechanism that assigns varying weights to positive and negative examples based on the distribution of classes.

Sadhukhan and Palit proposed the Lattice and Imbalance Informed Multi-Label Learning (LIIML) technique [22]. The process consists of two primary stages: choosing features and managing class imbalance. First, by obtaining each label's intrinsic positive and negative class lattices, LIIML retrieves features for each label.

Second, it adopts a misclassification cost method to tackle class imbalance. LIIML is considered a hybrid method, integrating both approaches, problem transformation in the first

step and algorithm adaptation in the second. The last approach is the Ensemble of Multi-Label Classification (EMLC), involving several classifiers for better performance.

Sun et al. [23] have developed an efficient ensemble paradigm named Multi-Label Ensemble with Adaptive Windowing (MLAW). Their algorithm employs Jensen–Shannon divergence to detect different kinds of concept drift. Another advantage of this method is that it prunes away infrequent label combinations to improve classification performance.

Moreover, the ensemble can also be used with the algorithm adaptation approach. MLSAkNN is also a base classifier in the Adaptive Ensemble of Self-Adjusting Nearest Neighbor Subspaces (AESAKNNS) algorithm proposed in [24].

It uses Poisson distribution to distribute a unique subset of features and samples for training to each classifier within the ensemble. The ADWIN detector also monitors classifiers. Li et al. [25] have introduced an effective classification of multi-label data streams with high-dimensional attributes and concept drifts.

The proposed algorithm utilizes an ensemble of classifiers and includes a feature selection and concept drift detection based on the max-relevance and min-redundancy approach. Law and Ghosh [26] have proposed a Multi-Label Binary Tree of Classifiers (ML-BTC) that maintains the label dependencies and class imbalance.

ML-BTC aims to construct a tree of classifiers for multi-label classification with a novel label-space partitioning method. During the training phase, the tree grows, and the decision at any node depends on the multi-label entropy and sample cardinality. Also, the unnecessary branching of imbalanced classes is restricted.

The final label set is assigned using the specific classifiers at the tree leaf nodes. Wu et al. [27] have proposed a weighted ensemble for multi-label classification called the Weighted Ensemble classification algorithm based on the Nearest Neighbors for Multi-Label data stream (WENNML).

An Active Candidate Ensemble Classifier is trained using a data stream block to generate the classifier and monitored by the well-known drift detector, ADWIN. All the detected instances will be stored in a data block, where a threshold sets its size. If the number of instances exceeds the threshold, a Passive Candidate Ensemble Classifier will be trained over the data in the block to generate a classifier; otherwise, no training is conducted. Both generated classifiers are dynamically updated using geometric and weighting techniques, replacing the old classifiers with more representative classifiers.

### 4.2 Q2: What Multi-label Classification Methods were Proposed ForIoT Streams?

IoT is one of the main data stream sources; however, not all the methods found have been evaluated on a streaming IoT in a non-stationary environment. Flood monitoring is one of the event monitoring applications in smart cities. Real-time monitoring of gully and drainage blockage is needed to avoid the flood.

Thus, Mishra et al. [28] have proposed an image-cropping-based DCNN model. The main task of the proposed model is to classify blockage images according to their severity. It has been trained over a dataset of 1200 images, drains, and gullies collected by the authors from Google images and YouTube videos.

Anomaly detection is a popular research field; it plays a vital role in cyber security for smart city services. Anomaly detection methods were considered binary classification problems with normal and abnormal classes.

However, such methods cannot always satisfy the demands of the smart home. Xu et al. [29] have proposed an algorithm to leverage the data produced by IoT services for anomaly detection considering the concept drift phenomenon, named Improved Long Short-Term Memory (I-LSTM).

It detects anomalies for IoT services using a Recurrent Neural Network (RNN) and then classifies the anomaly's specific category, which can belong to more than one category. It also adapts the drifts to increase detection accuracy effectively.

The authors used a real communication dataset collected from the IoT environment, which contains the communication between intelligent devices in the smart home. The Internet of Things has changed how manufacturing machinery works, as several machines can communicate and transfer data. Dalle Pezze et al. [30] have proposed a

method called Balance Among Tasks Optimizing Class Distribution in Memory ( BAT-OCDM).

The proposed framework is based on multi-label classification for packaging equipment monitoring. BAT-OCDM has been validated on the industrial Alarm Forecasting task originating from monitoring packaging equipment, which mainly uses the alarm logs for predictive maintenance purposes. The experimental results can effectively handle the presence of class imbalance and the distribution shifts in a stream of machines.

Human activity recognition has gained increased scientific interest due to the necessity of monitoring the multi-resident daily activities concurrently acting in smart homes. To deal with the multi-resident human activity recognition problem, Jethanandani et al. [31] have used the Classifier Chain model (CC) to predict human activities in smart homes.

The model has been applied to a real dataset of human activities in two houses with multiple residents. The model has been implemented using four classifiers, including Bernoulli Naïve Bayes, Decision Tree, K-nearest Neighbors, and Logistic Regression as base classifiers.

The results of the experiments showed that the developed Classifier Chain model with a Decision Tree outperformed the others and successfully dealt with the issue of recognizing multi-resident activities in a smart home. Similarly, Li et al.[32] have adopted a multi-label Markov Logic Network classification method (Ml-MLN) to recognize different individuals. Ml-MLN can identify resident types in multi-resident family homes based on their daily activity patterns and preferences.

Using Internet of Things technology, human motion detection systems have been developed to track and monitor people suffering from mobility difficulties. Wang et al. [33] have proposed a real-time walking motion detection system based on a multi-label imbalanced classification method, mobile phone, and motion stick.

For MLC, the authors have introduced MFGBoost, a combination of focal loss and LightGBM, mainly to classify human motions. Multi-label classification and IoT have also been used to improve the tourist experience in smart cities. Cepeda-Pacheco and Domingo have introduced a multi-label classifier to suggest the places, activities, or attractions that fit the tourists' profiles in real-time.

A deep neural network is trained with tourist profiles, and the data is collected using IoT devices in a smart city, including the visited attractions, location, time, and weather. Several sites are predicted based on the number of staying days, taking into account the nearest locations and the weather forecast.

Web Application Programming Interfaces (APIs) are well-known for providing services in Internet-of-Things (IoT) ecosystems. One of the main features of Web APIs is annotation or tagging, which plays a vital role in managing a huge number of services.

Xu et al. [34] have proposed a holistic framework utilizing multi-label classification for automatically tagging mobile and edge services in IoT systems. The framework employs neural network models based on a novel multi-head self-attention mechanism.

Such a mechanism learns the hidden correlation among annotations. Similarly, in things categorization, things are annotated with semantic labels, which can be used for searching and recommendation purposes. Chen et al. [35] have developed a novel method utilizing a binary support vector machine classifier for each label. It can produce semantic labels for a given thing.

Non-intrusive load monitoring is a technique used to monitor the total energy consumption of a building. With the emergence of IoT devices, handling massive data in such systems is challenging. Nalmpantis and Vrakas [36] have proposed a new framework tackling the power disaggregation issue called multi-label NILM.

The framework uses the dimensionality reduction method and addresses the disaggregation problem by identifying many appliances. The embedded microcontroller unit (MCU) is a vital component in supporting real-time processing in IoT.

Programmers usually face many challenges in developing code for microcontroller units and searching for sample codes online. Thus, Zhou et al. [37] proposed a tag-correlated-based classifier to help programmers query desired codes using tags. It has two machine-learning channels for processing the code description and analyzing the code content.

**Table 1.** Versatile multi-label classification methods

| Algorithm | Objectives | Address concept drift? | Address class imbalance? | Reference |
|---|---|---|---|---|
| **WENNML (2023)** | To detect drifting data during multi-label classification. | ✓ | X | [26] |
| **High-Dimensional MLC (2023)** | To detect drift and select optimal features for multi-label classification. | ✓ | X | [24] |
| **ML-INC (2023)** | Toaddressclassimbalance, incompletelabels, irrelevantfeatures, and noisy data in MLC. | X | ✓ | [17] |
| **BAT-OCDM (2023)** | Utilizemulti-labelclassification in thepackagingindustry. | ✓ | ✓ | [29] |
| **AESAKNNS (2022)** | To propose an ensemble MLC method for drifting data stream. | ✓ | X | [23] |
| **Semantic Annotation (2022)** | Togenerateannotationsthathelpdeveloperschoosethe appropriate Web APIs, facilitatingthedevelopment of services in IoTsystems. | X | X | [33] |
| **IMLSF (2022)** | Totackleclassimbalanceissues in multi-labelclassificationbyassigningweightstolabelled and unlabeledinstances. | ✓ | X | [20] |
| **MFGBoost (2022)** | Toclassify human motions in a real-time walkingmotiondetectionsystem. | X | ✓ | [32] |
| **ML-BTC (2022)** | To enable an ensemble of tree classifiers to preserve the label dependencies and handle the class imbalance in MLC. | X | ✓ | [25] |
| **Tourist Attraction RS (2022)** | To predict sites that fit tourists in a smart city leveraging IoT devices. | X | X | [1] |
| **ML-KELM (2022)** | To apply multi-label classification on the data stream in SIoT. | ✓ | X | [37] |
| **MLSAkNN (2021)** | To predict the complete set of labels from the constantly changing stream. | ✓ | X | [19] |
| **A tag correlated with MLC (2021)** | To assist embedded programmers in finding microcontroller unit codes. | X | X | [36] |
| **IC-based DCNN (2020)** | To monitor floods in smart cities using MLC of gullies and drainage images. | X | X | [27] |
| **I-LSTM (2020)** | To classify the anomaly category in the IoT environment. | ✓ | ✓ | [28] |
| **CC (2020)** | To predict human activities in smart homes with multi-residents. | X | X | [38] |
| **Things categorization (2020)** | To predict the labels of a given thing. | X | X | [34] |
| **MI-MLN (2020)** | To recognize humans in multi-resident homes. | X | X | [31] |
| **ML-NILM (2020)** | To enable the NILM framework to identify many appliances. | X | X | [35] |
| **LIIML(2020)** | To address class imbalance in multi-label datasets. | X | ✓ | [21] |
| **ML-CIB (2019)** | To address multi-label data issues such as class imbalance, label correlation, and irrelevant features | X | ✓ | [15] |
| **MLAW (2019)** | To propose a classification framework that predicts the class of the new incoming instance. It also tackles concept drift. | ✓ | X | [22] |
| **MLC via label correlation (2019)** | To predict a set of labels of a stream instances and tackle the concept drift. | ✓ | X | [14] |
| **MLSAMPkNN (2019)** | To propose a versatile MLC for drifting data streams for diverse learning scenarios | ✓ | ✓ | [18] |
| **MLC based on assessments of cost and value (2018)** | To solve the problem of class imbalance in MLC. | X | ✓ | [16] |

It also addresses the correlation between them to determine which function modules will mostly be used for a specific application.

The proposed method employs seven well-known classifiers, such as Binary Relevance, ML-KNN, and Classifier Chains. In addition, it was evaluated using an embedded code dataset built by the researchers. The idea of the Social Internet of Things (SIoT) is formed by integrating the Internet of Things with social networking platforms.

Information can be disseminated more quickly when multi-label categorization in SIoT provides multi-dimensional search terms for an object. For example, when a user uses labels or keywords to query for an object, the object content can be sent from the source to the user who is interested in it.

Nevertheless, this procedure is more difficult in a data stream environment than in a stationary setting. Working in a streaming setting, Luo et al. [38] have developed a multi-label approach based on Kernel Extreme Learning Machine (ML-KELM).

The adaptive threshold and kernel extreme learning machine form the foundation of ML-KELM. It makes use of the Sherman-Morrison-Woodbury algorithm and the Cholesky decomposition method. Additionally, ML-KELM can adjust to concept drift in the data stream.

### 4.3 Q3: Do Multi-label Classification Methods in the Literature Address Concept Drift and Class Imbalance?

Most surveyed studies have addressed concept drift in MLC tasks in data streams. The few that have not addressed them are all in the context of IoT. On the other hand, class imbalance has not been studied enough when dealing with MLC tasks in data streams generally and in the context of IoT. Table 1 shows the multi-label classification methods in the literature arranged by the dates from the most recent to the earliest. It also presents their objectives and whether class imbalance and concept drift were addressed

### 4.4 Q4: What Are The Main Multi-label Classification Datasets and Tools?

Datasets and software implementation of the MLC algorithms are required to carry out a benchmarking comparison of several methods [39]. The datasets adopted in the literature to evaluate the performance of the algorithms in multi-label streaming environments are classified into two types:

- **Real-world Datasets:** These are collected by authors and correspond to real-world applications. Examples of real-world datasets for MLC are listed below:
- **ARAS (IoT dataset):** Is an available online dataset consisting of truth labels for 27 human activities collected from real people in real houses [40]. The two houses have seven ambient sensors, with two residents in each. Residents did not follow any specific scenario during the data collection, which took two months.
- **Sensor (IoT dataset):** is a dataset collected using a TinyDB in-network query processing system [41]. Fifty-four sensors were deployed in the Intel Berkeley Research lab over 37 days. The dataset includes four attributes: humidity, temperature, light, and voltage, and one class isdivided into four labels.
- **MediaMill: C**onsists of 43907 data instances of video frames; each has 120 features and 101 class labels . The main task is to annotate video frames with the correct class labels semantically.
- **IMDB**: Is a dataset of 120,919 records of movie plot text summaries. It is employed for text classification, where each record has 1,001 attributes and 28 class labels.
- **Ohsumed:** Is a dataset of 2,417 records of peer-reviewed medical articles. It has 103 attributes and 23 class labels, each representing disease categories.

**Synthetic Datasets: generators from software tools produce artificial dataset**s: Such datasets are useful as they provide the ground truth of the data. Unlike real datasets, drift detection could be evaluated in synthetic datasets since the drift location, type, and duration are determined.

- **Waveform Generator:** Consists of three classes and 40 numeric attributes; each class is formed by combining two of three basewaves [6]. Concept drifts can be added by exchanging the positions of the attributes,

where each attribute represents a particular context.

- **Random Tree:** Generates a data stream based on a randomly generated tree [42]. It also allows for customizing the number of attributes and classes. The tree is developed by selecting the attributes as split nodes and assigning classes to them randomly. Concept drift can be added by changing the parameters, such as the average number of labels per sample or the label dependencies. Moreover, there are a lot of open-source tools or libraries for machine learning, which provide many methods, including classification, regression, clustering, and many others. These tools also allow the researchers to implement and evaluate their proposed algorithms directly.
- **Massive Online Analysis (MOA):** Is a well-known framework for data stream learning [43]. MOA is based on the WEKA library and written in Java. It provides various popular machine learning algorithms, MLC algorithms and data stream generators. MOA also allows drift simulation and provides several evaluation methods.
- **Scikit-multiflow:** Is an open-source framework for learning from data streams in Python. [44]. It extends the well-known sci-kit-learn designed to accommodate several data stream algorithms. Scikit-multiflow provides several stream generators, learners, evaluators, and popular change detectors.
- **Mulan is an Open-source library developed in Java that applies to** multi-label learning. It contains several state-of-the-art algorithms concerning multi-label learning. Mulan also provides evaluation measures by hold-out and cross-validation.
- **MEKA:** is a framework built on WEKA, a machine learning tool . It provides several multi-label learning methods.
- **Hadoop SPARK:** is a software designed to process large-scale data supporting many languages such as Python, Java, Scala, and R [45]. It involves a built-in library that includes SQL, Spark Streaming, MLlip, and GraphX. MLlip is a machine-learning library containing many algorithms for classification, regression, clustering, association rules, etc.

## 5 Discussion

Smart systems generate data streams in varying forms, at high speeds, and in massive volumes. Most traditional machine learning algorithms assume that each instance is associated with only a single class label, which is not true in many real-world cases. The methods under versatile multi-label classification have been evaluated regarding music/ image categorization, text categorization, bioinformatics, and many others [46].

Most studies discovered different types of data changes that occur suddenly or gradually. However, the class imbalance was not extensively inspected, although it is a well-known challenge in real-world data stream classification problems. Moreover, many methods showed a high complexity and long classification time.

The Internet of Things (IoT) is an expanding and growing field that has attracted researcher attention in the last decade. Methods explored in the literature showed the diversity of IoT applications and the importance of multi-label classification, specifically in smart cities. MLC methods can monitor floods in smart cities utilizing gully and drainage images [47]. They also can predict sites that fit tourists by leveraging IoT devices. Besides, MLC methods are used to recognize several activities of residents in smart homes [48].

However, most studies did consider drift detection and class imbalance. Drift is one of the main characteristics of the data stream, and class imbalance is very common in the IoT data stream. Therefore, MLC models leveraging IoT data stream must tackle these challenges to avoid wrong predictions.

Despite the extensive research on the MLC of data stream, the joint treatment of concept drift and class imbalance in the context of IoT or data stream is still not largely explored. Furthermore, concept evolution is another MLC challenge that could be further researched.

It occurs when completely new labels appear where they have never emerged before. Features also evolve; known features might fade, and new

ones emerge. Still, existing MLC methods do not significantly recognize the evolution of the stream's class labels and features.

## 6 Conclusion and Future Works

Data stream Classification has many challenges, including concept drift, class imbalance, and multi-label data. In many real-world applications, data instances are associated with many labels, where single classification models are unsuitable.

Also, classification models without a drift detector won't be able to capture new data distribution; therefore,they will keep learning from prior training data, resulting in inaccurate predictions. Furthermore, classification models should be able to deal with imbalanced data to avoid bias toward majority classes and deteriorate the model's performance.

This article reviews the state-of-the-art versatile multi-label classification. It also provides the MLC methods developed for the IoT stream. However, some methods have not addressed drift detection, especially those developed for IoT.

Furthermore, most studies did not consider the data imbalance. The joint treatment for data imbalance, multi-label, and concept drift remains unexplored.Future works of IoT data streams can be integrated with edge computing, predictive analytics and data fusion techniques for effective system monitoring.

## References

1. **Cepeda-Pacheco, J. C., Domingo, M. C. (2022).** Deep learning and internet of things for tourist attraction recommendations in smart cities. Neural Computing and Applications, Vol. 34, No. 10, pp. 7691–7709. DOI: 10.1007/s00521-021-06872-0.
2. **Nguyen, H., Woon, Y., Ng, W. (2014).** A survey on data stream clustering and classification. Knowledge and Information Systems, Vol. 45, No. 3, pp. 535–569. DOI: 10.1007/s10115-014-0808-1.
3. **Hurbean, L., Danaiata, D., Militaru, F., Dodea, A., Negovan, A. (2021).** Open data based machine learning applications in smart cities: a systematic literature review. Electronics, Vol. 10, No. 23, pp. 2997. DOI: 10.3390/electronics10232997.
4. **Rivera, G., Florencia, R., García, V., Ruiz, A., Sánchez-Solís, J. P. (2020).** News classification for identifying traffic incident points in a spanish-speaking country: a real-world case study of class imbalance learning. Applied Sciences, Vol. 10, No. 18, pp. 6253. DOI: 10.3390/app10186253.
5. **Zheng, X., Li, P., Chu, Z., Hu, X. (2020).** A survey on multi-label data stream classification. IEEE Access, Vol. 8, pp. 1249–1275. DOI: 10.1109/access.2019.2962059.
6. **Endut, N., Hamzah, W. M. A. F., Ismail, I., Kamir-Yusof, M., Abu-Baker, Y., Yusoff, H. (2022).** A systematic literature review on multi-label classification based on machine learning algorithms. TEM Journal, pp. 658–666. DOI: 10.18421/tem112-20.
7. **Herrera, F., Charte, F., Rivera, A. J., del-Jesus, M. J. (2016).** Multilabel classification. Springer International Publishing. DOI: 10.10 07/978-3-319-41111-8.
8. **Sanghi, G., Kanungo, N., Deshmukh, S., Agarwal, S. (2017).** Automatic multi-label image annotation for smart cities. IEEE Region 10 Symposium, Vol. 14, pp. 1–4. DOI: 10.110 9/tenconspring.2017.8069997.
9. **Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A. (2014).** A survey on concept drift adaptation. ACM Computing Surveys, Vol. 46, No. 4, pp. 1–37. DOI: 10.11 45/2523813.
10. **Wang, P., Jin, N., Fehringer, G. (2020).** Concept drift detection with false positive rate for multi-label classification in IoT data stream. International Conference on UK-China Emerging Technologies, Vol. 11, pp. 1–4. DOI: 10.1109/ucet51115.2020.9205421.
11. **García, V., Sánchez, J., Marqués, A., Florencia, R., Rivera, G. (2020).** Understanding the apparent superiority of over-sampling through an analysis of local information for class-imbalanced data. Expert Systems with Applications, Vol. 158, pp. 113026. DOI: 10.1016/j.eswa.2019.113026.

12. **Bolívar, A., García, V., Florencia, R., Alejo, R., Rivera, G., Sánchez-Solís, J. P. (2022).** A preliminary study of smote on imbalanced big datasets when dealing with sparse and dense high dimensionality. Pattern Recognition, Vol. 13264, pp. 46–55. DOI: 10.1007/978-3-031-07750-0_5.
13. **Wolfswinkel, J. F., Furtmueller, E., Wilderom, C. P. M. (2013).** Using grounded theory as a method for rigorously reviewing literature. European Journal of Information Systems, Vol. 22, No. 1, pp. 45–55. DOI: 10.1057/ejis.2011.51.
14. **Nguyen, T. T., Nguyen, T. T. T., Luong, A. V., Nguyen, Q. V. H., Liew, A. W., Stantic, B. (2019).** Multi-label classification via label correlation and first order feature dependance in a data stream. Pattern Recognition, Vol. 90, pp. 35–51. DOI: 10.1016/j.patcog.2019.01.007.
15. **Braytee, A., Liu, W., Anaissi, A., Kennedy, P. J. (2019).** Correlated multi-label classification with incomplete label space and class imbalance. ACM Transactions on Intelligent Systems and Technology, Vol. 10, No. 5, pp. 1–26. DOI: 10.1145/3342512.
16. **Ding, M., Yang, Y., Lan, Z. (2018).** Multi-label imbalanced classification based on assessments of cost and value. Applied Intelligence, Vol. 48, No. 10, pp. 3577–3590. DOI: 10.1007/s10489-018-1156-8.
17. **Ji, X., Tan, A., Wu, W., Gu, S. (2023).** Multi-label classification with weak labels by learning label correlation and label regularization. Applied Intelligence, Vol. 53, No. 17, pp. 20110–20133. DOI: 10.1007/s10489-023-04562-z.
18. **Roseberry, M., Krawczyk, B., Cano, A. (2019).** Multi-label punitive kNN with self-adjusting memory for drifting data streams. ACM Transactions on Knowledge Discovery from Data, Vol. 13, No. 6, pp. 1–31. DOI: 10.1145/3363573.
19. **Roseberry, M., Krawczyk, B., Djenouri, Y., Cano, A. (2021).** Self-adjusting k nearest neighbors for continual learning from multi-label drifting data streams. Neurocomputing, Vol. 442, pp. 10–25. DOI: 10.1016/j.neucom.2021.02.032.
20. **Rastogi, R., Mortaza, S. (2022).** Imbalance multi-label data learning with label specific features. Neurocomputing, Vol. 513, pp. 395–408. DOI: 10.1016/j.neucom.2022.09.085.
21. **Sadhukhan, P., Palit, S. (2020).** Lattice and imbalance informed multi-label learning. IEEE Access, Vol. 8, pp. 7394–7407. DOI: 10.1109/access.2019.2962201.
22. **Sun, Y., Shao, H., Wang, S. (2019).** Efficient ensemble classification for multi-label data streams with concept drift. Information, Vol. 10, No. 5, pp. 158. DOI: 10.3390/info10050158.
23. **Alberghini, G., Barbon-Junior, S., Cano, A. (2022).** Adaptive ensemble of self-adjusting nearest neighbor subspaces for multi-label drifting data streams. Neurocomputing, Vol. 481, pp. 228–248. DOI: 10.1016/j.neucom.2022.01.075.
24. **Li, P., Zhang, H., Hu, X., Wu, X. (2022).** High-dimensional multi-label data stream classification with concept drifting detection. IEEE Transactions on Knowledge and Data Engineering, Vol. 35, No. 8, pp. 8085–8099. DOI: 10.1109/tkde.2022.3200068.
25. **Law, A., Ghosh, A. (2022).** Multi-label classification using binary tree of classifiers. IEEE Transactions on Emerging Topics in Computational Intelligence, Vol. 6, No. 3, pp. 677–689. DOI: 10.1109/tetci.2021.3 075717.
26. **Wu, H., Han, M., Chen, Z., Li, M., Zhang, X. (2023).** A weighted ensemble classification algorithm based on nearest neighbors for multi-label data stream. ACM Transactions on Knowledge Discovery from Data, Vol. 17, No. 5, pp. 1–21. DOI: 10.1145/3570960.
27. **Mishra, B. K., Thakker, D., Mazumdar, S., Neagu, D., Gheorghe, M., Simpson, S. (2020).** A novel application of deep learning with image cropping: a smart city use case for flood monitoring. Journal of Reliable Intelligent Environments, Vol. 6, No. 1, pp. 51–61. DOI: 10.1007/s40860-020-00099-x.
28. **Xu, R., Cheng, Y., Liu, Z., Xie, Y., Yang, Y. (2020).** Improved long short-term memory based anomaly detection with concept drift adaptive method for supporting IoT services. Future Generation Computer Systems, Vol.

112, pp. 228–242. DOI: 10.1016/j.future.2020.05.035.

29. **Dalle-Pezze, D., Deronjic, D., Masiero, C., Tosato, D., Beghi, A., Susto, G. A. (2023).** A multi-label continual learning framework to scale deep learning approaches for packaging equipment monitoring. Engineering Applications of Artificial Intelligence, Vol. 124, pp. 106610. DOI: 10.1016/j.engappai.2023.106610.
30. **Jethanandani, M., Sharma, A., Perumal, T., Chang, J. (2020).** Multi-label classification based ensemble learning for human activity recognition in smart home. Internet of Things, Vol. 12, pp. 100324. DOI: 10.1016/j.iot.2020.100324.
31. **Li, Q., Huangfu, W., Farha, F., Zhu, T., Yang, S., Chen, L., Ning, H. (2020).** Multi-resident type recognition based on ambient sensors activity. Future Generation Computer Systems, Vol. 112, pp. 108–115. DOI: 10.1016/j.future.2020.04.039.
32. **Wang, J., Jiang, X., Meng, Q., Saada, M., Cai, H. (2022).** Walking motion real-time detection method based on walking stick, IoT, copod and improved lightgbm. Applied Intelligence, Vol. 52, No. 14, pp. 16398–16416. DOI: 10.1007/s10489-022-03264-2.
33. **Xu, Y., Xiao, W., Yang, X., Li, R., Yin, Y., Jiang, Z. (2023).** Towards effective semantic annotation for mobile and edge services for internet-of-things ecosystems. Future Generation Computer Systems, Vol. 139, pp. 64–73. DOI: 10.1016/j.future.2022.09.021.
34. **Chen, Y., Zhang, J., Xu, L., Guo, M., Cao, J. (2020).** Modeling latent relation to boost things categorization service. IEEE Transactions on Services Computing, Vol. 13, No. 5, pp. 915–929. DOI: 10.1109/tsc.2017.2715159.
35. **Nalmpantis, C., Vrakas, D. (2020).** On time series representations for multi-label NILM. Neural Computing and Applications, Vol. 32, No. 23, pp. 17275–17290. DOI: 10.1007/s00521-020-04916-5.
36. **Zhou, Y., Cui, S., Wang, Y. (2021).** Machine learning based embedded code multi-label classification. IEEE Access, Vol. 9, pp. 150187–150200. DOI: 10.1109/access.2021.3123498.
37. **Luo, F., Liu, G., Guo, W., Chen, G., Xiong, N. (2022).** ML-KELM: A kernel extreme learning machine scheme for multi-label classification of real time data stream in SIoT. IEEE Transactions on Network Science and Engineering, Vol. 9, No. 3, pp. 1044–1055. DOI: 10.1109/tnse.2021.3073431.
38. **Kasubi, J. W., Huchaiah, M. D. (2021).** Human activity recognition for multi-label classification in smart homes using ensemble methods. Artificial Intelligence and Sustainable Computing for Smart City, Communications in Computer and Information Science, Vol. 124, pp. 282–294. DOI: 10.1007/978-3-030-82322-1_21.
39. **Madden, S. (2022).** Intel Lab Data. db.csail.mit.edu/labdata/labdata.html
40. **Museba, T., Nelwamondo, F., Ouahada, K. (2021).** An adaptive heterogeneous online learning ensemble classifier for nonstationary environments. Computational Intelligence and Neuroscience, Vol. 2021, No. 1. DOI: 10.1155/2021/6669706.
41. **Bifet, A., Read, J., Holmes, G., Pfahringer, B. (2018).** Streaming data mining with massive online analytics (MOA). Series in Machine Perception and Artificial Intelligence, Data Mining in Time Series and Streaming Databases, pp. 1–25. DOI: 10.1142/9789813228047_0001.
42. **Montiel, J., Read, J., Bifet, A., Abdessalem, T. (2018).** Scikit-multiflow: A multi-output streaming framework. Journal of Machine Learning Research, Vol. 19, No. 72, pp. 1–5.
43. **Apache SparkTM (2022).** Unified engine for large-scale data analytics. spark.apache.org/
44. **Ganji, R. N., Dadkhah, C., Tohidi, N. (2023).** Improving sentiment classification for hotel recommender system through deep learning and data balancing. Computación y Sistemas, Vol. 27, No. 3, pp. 811–825. DOI: 10.13053/cys-27-3-4655.
45. **Moreno-Escobar, J. J., Pérez-Franco, V. J., Coria-Páez, A. L., Morales-Matamoros, O., Aguilar-del-Villar, E. Y., Castillo-Perez, M. D. (2023).** Multivariate data analysis of

consumer behavior of functional products: a neuroscience and neuromarketing approach to improve decision-making. Computación y Sistemas, Vol. 27, No. 4, pp. 1027–1046. DOI: 10.13053/cys-27-4-4690.

46. **Gelbukh, A., Pérez-Alvarez, D. A., Kolesnikova, O., Chanona-Hernandez, L., Sidorov, G. (2024).** Multi-instrument based n-grams for composer classification task. Computación y Sistemas, Vol. 28, No. 1, pp. 85–98. DOI: 10.13053/cys-28-1-4903.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Mashail Althabiti.*

ISSN 2007-9737

# HELI: An Ensemble Forecasting Approach for Temperature Prediction in the Context of Climate Change

Erick Estrada-Patiño[1], Guadalupe Castilla-Valdez[1], Juan Frausto-Solis[1,*], Juan Javier Gonzalez-Barbosa[1], Juan Paulo Sánchez-Hernández[2]

[1] Tecnológico Nacional de México,
Instituto Tecnológico de Ciudad Madero,
Mexico

[2] Universidad Politécnica del Estado de Morelos,
Information of Technology Division,
Mexico

estrada1792@gmail.com, gpe_cas@yahoo.com.mx,
jjgonzalezbarbosa@hotmail.com, juan.paulosh@upemor.edu.mx

**Abstract.** Climate change is a critical challenge, demanding the development of effective methods for temperature forecasting. Statistical and machine learning models emerge as promising alternatives. However, there is no widely accepted superior method; ensemble approaches integrate strategies that take advantage of each forecasting method. Ensemble methodologies combine methods, weighing their participation to integrate each of them. Forecasting researchers have shown that evolutionary algorithms are highly effective in achieving an ensemble that is at least as effective as the best single method. This paper presents HELI, a forecasting methodology designed to forecast the climate temperature variable; its architecture is modular, aiming to provide a flexible forecasting application in the climate change area. We present experimentation and a hypothesis test for a region in Mexico City and show HELI's competitiveness compared to leading strategies. Besides, we present experiments with other climate change variables that show HELI flexibility in the context of climate change.

**Keywords.** Ensemble methods, LSTM, CNN, evolutive ponderation.

## 1 Introduction

The current climate change phenomena are crucial problems that require exhaustive study [1]. Understanding and correcting these dynamics is essential to anticipate and mitigate their possible long-term consequences [2].

Besides, the United Nations Framework Convention on Climate Change, specifically the 2015 Paris Agreement, proposed that global average temperature increase should not exceed 1.5 °C, with a critical threshold of 2°C by 2100 [3]. This convention defined temperature as a fundamental variable for estimating the evolution and impacts of climate change [4, 5, 6, 7].

Therefore, it is imperative to research strategies and tools for predicting temperatures beyond seasonal periods to cover the medium and long term [8, 9]. Although traditional climate models use current and past observations, they are not accurate methods due to anthropogenic, natural, and other factors associated with climate change. On the other hand, forecasting models, whether classical or based on machine learning algorithms, are a valuable alternative.

These approaches model nonlinear relationships among several climate change variables based on historical data. Their effectiveness in many areas, including weather forecasting in the context of climate change, has been demonstrated and equated with conventional meteorological strategies [10, 11, 12]. Many authors have explored climate phenomena using statistical strategies such as SARIMA and Exponential Smoothing, which capture complex behaviors with good performance [8, 10].In addition, neural networks such as LSTM and CNN,

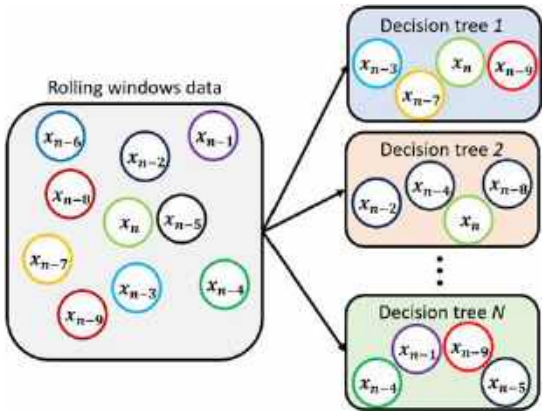



**Fig. 1.** Division of information into a random forest

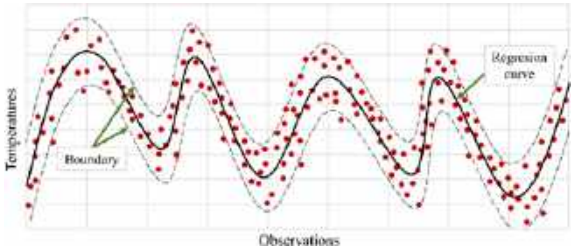



**Fig. 2.** Confidence intervals in support vector regression

Random Forest Regression (RFR), and Support Vector Regression have yielded competitive results [11, 12, 9]. The Forecasting and Machine Learning communities have observed that no method is superior to any other for all instances of a given problem in their areas [13, 14].

Given this difficulty of choice, hybrid and ensemble models have been proposed to enhance the forecast of a variable in any domain. This work proposes forecasting temperature; thus, we designed HELI to combine single forecasting methods with a relevant variable for climate change. Ensemble methods, in their general form, unify several strategies, taking advantage of the best of each one to deliver a high-quality integrated forecast [15, 16, 17].

These strategies have demonstrated success, highlighting the superior performance of SMYL in the M4 competition [18]. These are comparable to individual machine learning or classical strategies [18, 19]. In this paper, we present HELI, a methodology for temperature forecasting in the context of climate change. HELI incorporates robust preprocessing that facilitates the tuning and learning of multiple forecasting strategies, both classical and machine learning.

These strategies are combined and optimized for improved participation to produce a regression function that is at least as effective as the best single-trained method. The results evidence the competitiveness of HELI against leading state-of-the-art assembly forecasting approaches, with a low computational cost and high scalability and flexibility in adjusting the methods and their configuration in the assembly.

The structure of this paper is as follows. Section two presents a comprehensive review of the approaches and methodologies used and their advantages and disadvantages; we provide a solid context for the proposed methodology. Section three describes the HELI methodology in detail, including its main components, the configuration used in this work, and considerations for its replication.

The fourth section presents the results of a case study demonstrating the performance of HELI in temperature forecasting in Mexico; we decided to test HELI and present its forecasting results with other climate change variables, letting us evaluate its flexibility in scenarios different from the original. We discuss the implications of the results for the decision-making and future improvements. Finally, in section five we present the conclusions of this work.

## 2 Related Methods

In this paper, we explore two forecasting approaches applicable to time series: classical approaches and approaches supported by machine learning algorithms.

Classical forecasting approaches are strategies that base their predictive model on statistical methods, allowing the identification of possible trends and seasonal patterns in the data. These models are powerful because they generally involve tuning a few parameters and effectively adjusting to previously described behaviors. In addition, they have a low computational cost [20].

In contrast, forecasting strategies based on machine learning facilitate an exhaustive analysis of historical data, especially when these exhibit complex or highly nonlinear patterns [21]. These

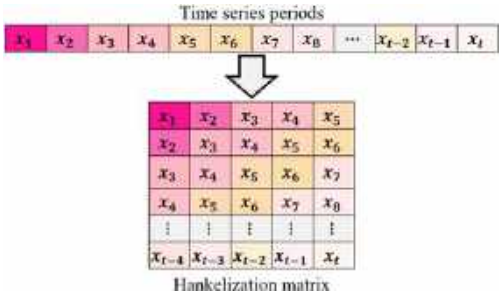



**Fig. 3.** Hankelization process

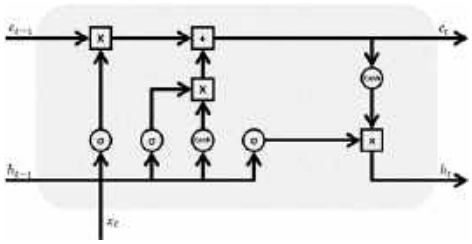


**Fig. 4.** LSTM cell

approaches have a high adaptive capacity and a remarkable tolerance to irregularities in the data; however, they usually entail a high computational cost, and the parameters to be adjusted are significantly more numerous compared to classical strategies [22].

Additionally, we explore strategies that evaluate the efficiency of algorithms beyond the improvement in error metrics on a data set [23]. Ranking methods, such as BIC or AIC, allow other metrics to be considered for scoring the effectiveness of an algorithm, with the understanding that a lower amount of error can mean an adequate strategy or an overfitted strategy for a given behavior [24].

These metrics make it possible to consider algorithms that present balanced results, translating into good performance in the long term [25]. Likewise, the weighting adjustment methods allow the solutions created by the ranking methods and some other random ones to evolve to find the best-weighted combination of the forecasting methods to issue a quality forecast.

## 2.1 Classical Forecasting Models

Classical forecasting methods are characterized by robust and solid strategies with statistical fundamentals, refined over an extensive period, enabling the development of robust models yielding competitive results [26].

These techniques are characterized by adjusting their hyperparameters based on historical data [27]. These methods maintain competitiveness when integrated in a hybrid mode with contemporary strategies, such as machine learning [18].

### 2.1.1 SARIMA

AutoRegressive Moving Average (ARMA) models have proven to be highly recognized and practical tools in the field of forecasting, dedicated to the analysis and prediction of time series [28]. Although ARMA focuses mainly on analyzing the autocorrelation and moving average present in the time series, it does not consider other significant factors that may manifest themselves in more complex series [29].

In this context, Seasonal Auto Regressive Integrated Moving Average (SARIMA) emerges as an evolution of ARIMA or an extension of ARMA. SARIMA addresses the limitations of ARMA by incorporating specific trend and seasonality components, thus allowing the generation of forecasts that adapt more efficiently to the behavior of the training data [30]:

$$\text{SARIMA}(p,d,q)(P,D,Q)s. \tag{1}$$

In general terms, SARIMA uses the orders specified in equation 1, where $p,d,q$ represent the orders for autoregressions, differences, and moving averages, respectively, applied to seasonally adjusted data. Likewise, $P,D,Q$ indicate the corresponding orders for autoregressions, differences, and moving averages, but this time applied to seasonal data.

The parameter $s$ represents the size of the seasonal window, a crucial component for capturing temporal patterns. SARIMA fitting is carried out using model selection techniques, such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC), as described in detail in later sections of this paper. These strategies provide a solid basis for model optimization, ensuring higher prediction accuracy and adaptability to the inherent complexities of the time series [31].

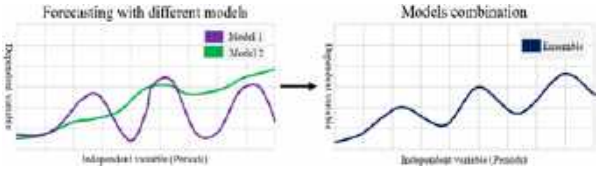



**Fig. 5.** Ensemble process

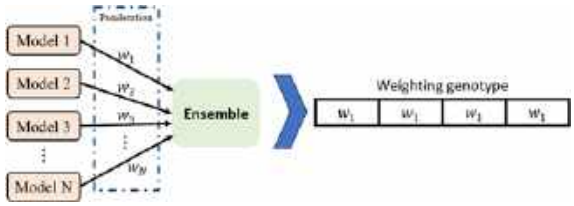



**Fig. 6.** Ponderation of forecasting models

### 2.1.2 Exponential Smoothing

Exponential Smoothing (ES) is an old and popular technique based on weighting the last data with the previous forecast; however, it is effective only for one period ahead. Holt & Winter is a valuable ES method, mainly combined with machine learning and ensemble methods [32].

This technique represents an advanced methodology commonly known as Holt&Winters-Triple (H&WT) in time series forecasting [33]. Derived from the simple exponential smoothing proposed by Brown & Holt, H&WT constitutes a sophisticated extension of double exponential smoothing [34].

In contrast to its predecessors, H&WT not only estimates a weighting that controls the influence of the current period in forecasting the next but also incorporates weights for the current trend and seasonality [35].

These weights are reflected in equations 2, 3, and 4, where the smoothing variables $\alpha$, ,$\beta$, $\gamma$, which require tuning for optimal adaptation to the behavior of the data, are highlighted.

The synthesis of these components is expressed in equation 5, which represents the expected observation for the next period:

$$A_t = \alpha(Y_t - S_{t-L}) + (1-\alpha)(A_{t-1} + T_{t-1}), \quad (2)$$

$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1}, \quad (3)$$

$$S_t = \gamma(Y_t - A_t) + (1-\gamma)S_{t-L}, \quad (4)$$

$$Y'_{t+p} = A_t + pT_t + S_{t-L+p}. \quad (5)$$

Despite the longevity of H&WT, it remains a highly effective strategy, evidenced by its accuracy in estimating seasonal components [18]. The consistent results it presents reinforce its continued relevance as a reliable tool in the time series forecasting landscape.

## 2.2 Machine Learning Forecasting Models

Machine learning methods represent artificial intelligence strategies dedicated to creating generalizations and learning patterns in data, which may be imperceptible to the human eye [22].

This achievement is materialized through the exhaustive analysis of data through statistical techniques that enable the identification of thresholds for the issuance of forecasts or decision-making. In this context, we address machine learning strategies specifically designed for time series forecasting in a general way [22].

These strategies focus on the ability to anticipate and model the evolution of data over time, thus improving forecasting accuracy in this context. This approach is essential to address problems related to the temporal dynamics of data and to maximize the usefulness of machine learning applications in the field of time series [32].

### 2.2.1 Random Forest

Random forest (RF) is one of the most effective decision tree strategies. RF builds an adaptive structure, defining and adjusting itself on a limited data set, this allows the trees to train with different sections of data and decreases the incidence of overfitting. As visualized in Figure 1, this structure exhibits reliability, allowing the adjustability of hyperparameters such as loss function, tree depth, and number of leaves.

However, the RF effectiveness may be bounded by high nonlinearity in the data or when the data are presented in large quantities [36, 37]. In response to these limitations, an effective technique is the integration of several decision trees operating on typically small data sets.

These techniques, known as bagging or boosting, provide superior results compared to individual decision trees [36, 38]. Random Forest Regression is positioned as a bagging technique that generates an abundance of decision trees [39]. These trees are trained with reduced

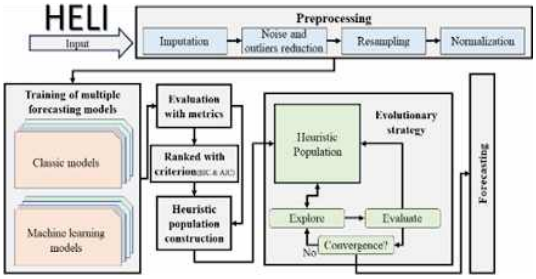



**Fig. 7.** HELI (Heuristic ensemble for learning information) methodology


**Fig. 8.** Data collection area

fragments of information, each emitting its forecast. Then, an ensemble criterion is implemented, following the philosophy of masses. In the regression area, an arithmetic mean is commonly described by $y = \frac{1}{T}\sum_{t=1}^{T} y_t$, where $y_t$ corresponds to the results of each decision tree. This ensemble approach offers notable advantages, such as its high parallelization capability, robustness, and adaptability to a wide variety of problems [40].

### 2.2.2 Support Vector Regression

SVR constructs a regression function aimed at minimizing error concerning the regression of the training data. The chosen kernel directly influences the efficacy of this function's fit. Additionally, two confidence intervals are established and depicted alongside the proposed regression curve.

These intervals are determined by the hyperparameters C and epsilon, which play a pivotal role in defining the flexibility and width of the tolerance margin [41]. Support Vector Machines (SVM) have demonstrated remarkable efficiency at a modest computational cost [42].

In this context, we delve into its variant suitable for time series: Support Vector Regression (SVR). The efficacy of SVR lies in its capacity to adapt to scenarios characterized by nonlinear behavior, a common occurrence in time series marked by significant noise levels or intricate seasonal patterns [41].

The SVR process is illustrated in Figure 2, wherein the SVM technique endeavors to optimize the fitting of the regression line to the data, accounting for the inherent complexity of the dataset, be it due to its nonlinear nature, presence of noise, or the existence of sophisticated seasonal patterns.

### 2.2.3 Convolutional Neural Networks

The convolutional neural network (CNN) methodology is a promising approach to the forecasting area, standing out for its ability to perform deep learning based on convolutional and pooling processes [43, 44].

Central to the model, these processes automatically identify relevant features in data sets [44]. This auto-generalization capability enables the model to discover significant patterns and features, which can be used to determine forecasts [8]. Traditionally, image recognition area involving matrix information successfully uses CNN models.

However, in the case of univariate time series, the representation takes a vector form. Figure 3 illustrates the Hankelization process, a technique that, by defining an amplitude value, generates a matrix representation of a time series [45]. The convolutional procedure in CNNs is deployed sequentially through one or more convolution and pooling layers.

The convolutional layers, fundamental in the architecture, apply convolution operations to the input data, enabling local pattern learning. In the specific context of time series forecasting,

**Algorithm 1.** General structure of methodology

**Algorithm 1** HELI

**Input** Time_series
**Output** Ensemble_ponderation, Error_metrics_results

**procedure** HELI
  $preprocessing\_ts \leftarrow preprocessing(time\_series)$
  $train, val, test \leftarrow timeSeriesSplit(preprocessing\_ts)$
  Define $models \leftarrow$ all_models_types
  **for each** model **in** models **do**
    $model.fit(train, val)$
    $mse\_models[model.index] \leftarrow calculateMSE(model)$
  **end for**
  $sol\_pop \leftarrow heuristicPopulation(models, mse\_models)$
  $best\_sol \leftarrow evolutiveStrategy(sol\_pop, models, train, val)$
  $solution \leftarrow evaluateEnsemble(best\_sol, test)$
  print($solution$)
**end procedure**

**Algorithm 2.** Generation of heuristic population

**Algorithm 2** Heuristic Population Generation

**Input** models, mse_models
**Output** Population

**procedure heuristicPopulation**($models, mse\_evaluations$)
  $population \leftarrow$ Empty
  $aic\_rnk\_w \leftarrow weightByPer(aicRnk(models, mse\_eval))$
  $population.append(aic\_rank\_w)$
  $bic\_rnk\_w \leftarrow weightByPer(bicRnk(models, mse\_eval))$
  $population.append(bic\_rank\_w)$
  $rmse\_w \leftarrow weightByPer(rmseRnk(models, mse\_eval))$
  $population.append(rmse\_w)$
  $uniform\_w \leftarrow weightByPer(uniPart(models))$
  $population.append(uniform\_w)$
  **for each** model **in** models **do**
    $weighting\_solution \leftarrow$ zeros()
    $weighting\_solution[models[model.index]] \leftarrow 100$
    $population.append(weighting\_solution)$
  **end for**
  for $n$ **in** rest_population **do**
    $solution \leftarrow randomFactibleSolution()$
    $population.append(solution)$
  **end for**
  **return** $population$
**end procedure**

convolutional layers are employed to discern temporal patterns in the data.

For example, a convolutional layer may specialize in identifying trend patterns, cycles, or seasonality. Pooling layers are incorporated to reduce the dimensionality of the data resulting from the convolutional layers. This process helps to optimize learning efficiency and mitigate the risk of overfitting. In the specific context of time series forecasting, convolutional layers are used to discern temporal patterns in the data.

Thus, a convolutional layer may specialize in identifying trend patterns, cycles, or seasonality. Pooling layers play a crucial role in reducing the amount of data to be processed, which is particularly beneficial for time series with many elements. After the convolutional phase, we apply internal vectorization to connect a multilayer perceptron (MLP) network.

In the forecast context, this network outputs a numerical value representing the forecast for a specific input. This connection process between the convolutional stage and the MLP culminates in generating the desired forecast.

### 2.2.4 Long Short-term Memory

This approach is based on recurrent neural networks, standing out for its ability to address various problems inherent to this structure [46, 47]. The introduction of cells represents a significant advance, substantially improving the forecasts' quality.

These cells, equipped with gates, play a crucial role in selecting which information to retain or discard in the short, medium, and even long term; this makes it possible to generate quality results in problems related to time series forecasting, enabling the generalization of short-term changes such as those of a seasonal or cyclical nature [48].

In contrast to traditional neural networks, which lack a structure that facilitates the relationship between sequences of previous and subsequent inputs, and recurrent networks that attenuate the influence of the past in large temporal data sets due to their nature, long short-term memory (LSTM) networks represent a functional alternative [8].

This model replaces single neurons with cells, which can be organized in layers and support stacking. This configuration contributes significantly to the learning of long-term dependencies. LSTM structure has four fundamental gates: the output gate, the input gate, the forgetting gate, and the memory gate.

These gates operate through activation functions, exercising precise control over the data flow, as illustrated in Figure 4. This level of control provided by the gates increases the model's predictive capability and confers essential flexibility to adapt to complex patterns and changes in the time series under study.

**Algorithm 3.** Evolutive process

**Algorithm 3** Evolutionary Strategy for Ensemble Optimization

**Input** Heuristic_population, Trained_models, Train, Val
**Output** Best_solution

**procedure evolutiveStrategy**(*H_pop, Trained_models, Train, Val*)
  Initialize evolutionary parameters
  $best\_solution \leftarrow Evaluate(H\_pop)$
  **while** not converged **do**
    $offspring\_pop \leftarrow$ GenOffspring(H_pop)
    $offspring\_pop \leftarrow mutation(offspring\_pop)$
    **for** each offspring in offspring_pop **do**
      $ensemble\_sol \leftarrow CombMdls(offspring, Trained\_models)$
      $ensemble\_mse \leftarrow Evaluate(ensemble\_sol, Train, Val)$
      **if** $ensemble\_mse < best\_solution\_mse$ **then**
        $best\_solution \leftarrow ensemble\_solution$
        $best\_solution\_mse \leftarrow ensemble\_mse$
      **end if**
    **end for**
    $H\_pop \leftarrow Update(H\_pop, offspring\_pop)$
  **end while**
  **return** *best_solution*
**end procedure**

## 2.3 Ensemble Methods

Forecasting strategies exhibit diverse behaviors, varying depending on the series processed and even on the specific segment of the series used to fit the model [15, 16]. The absence of a universally superior model stands out since the optimal performance of a model is not guaranteed in all cases or all segments of a series.

This well-recognized problem can be addressed by assembling different forecasting methods, thus generating a single result that incorporates the best individual features of each approach, either through hybridization or ensemble results [39].

Ensemble strategies are based on combining the results of two or more forecasting methods, assigning each of them a percentage of participation [15]. Figure 5 illustrates the essential idea of the assembly process by combining two or more strategies.

However, the simple combination or even the use of algorithm ranking metrics may, in most cases, not be optimal, as it does not match the specific behavior of the algorithms on the regression curves [49]. For this reason, weighting optimization methods emerge as an essential practice, ensuring results are at least as good as the best single method.

These methods perform an exhaustive exploration of the solution space, adjusting to the data and the algorithms participating in the ensemble. In this way, a precise and efficient adaptation to the inherent complexities of the time series is guaranteed, strengthening the robustness and accuracy of the forecasting process.

### 2.3.1 Evolutionary Weight Optimization

The weighting adjustment in an ensemble method constitutes a combinatorial optimization challenge, which may be infeasible to solve exhaustively. Therefore, the heuristic strategy obtains results in a reasonable time, and in this sense community scientists can use it to obtain fast results.

Evolutionary algorithms find their inspiration in biological evolution, where different individuals intermingle or mutate over generations, acquiring or improving skills necessary for their survival and discarding those not necessary or counterproductive [50].

In this strategy, each weighting or solution is treated as an individual, coding it to obtain its genotype and evaluating its performance to determine its fitness value, as shown in Figure 6. The evolutionary weighting strategy is derived from Holland's approach, which uses a population consisting of a set of solutions that will undergo an iterative evolutionary mechanism similar to biological generations [51].

This process selects parents by evaluating their fitness, and then a combination strategy of the parents produces offspring solutions. The replacement process applies a criterion based on the evaluation of the fitness to integrate into the next generation.

Between the process of generating new solutions and their evaluation for integration into the population, there is the possibility of modifying these new solutions by perturbing some genes of the solution; this strategy is known as mutation and allows greater exploration of the solution space, also decreasing the stagnation of the population and facilitating the rapid attainment of the convergence criterion.

The evolutionary strategy will evolve over a fixed number of generations or may end earlier or be extended, depending on the definition of different convergence criteria. These criteria evaluate the performance of the best solutions for

### 2.4 Rank Methods

In machine learning, prediction models are nurtured by strategies that seek to minimize a specific error metric. This metric commonly contrasts the predicted value with the actual value in a given period, thus accumulating errors throughout the training process. Several error metrics are available, each focusing on different aspects of the dispersion for the original value.

However, these metrics only guarantee the convergence of the forecast models to an optimal fit for the training and validation set. An inherent danger in a soon convergence is that it can lead to model overfitting, especially in strategies involving many parameters and hyperparameters [35].

On the other hand, ranking methods take a different approach when assessing the quality of a model. These methods integrate additional features along with the error metric to assign a score that more holistically reflects the effectiveness of the model.

This approach allows a balance to be stuck between model accuracy, computational complexity, run times, and other relevant factors. In essence, ranking methods offer a more comprehensive perspective for evaluating and selecting models, overcoming the limitations inherent in pure convergence to the training set fit.

#### 2.4.1 Akaike Information Criterion

The Akaike information criterion (AIC) is a fundamental statistical tool for evaluating forecasting methods [52]. Its purpose is to strike a balance between model accuracy, as measured by a likelihood function, and model complexity, which is mainly determined by the number of hyperparameters fitted to perform the forecast [53].

The value of the AIC decreases as the models achieves better results. This behavior is reflected in equation 6, where $k$ represents the number of hyperparameters of the model, and $L$ denotes the likelihood function. In this context, the likelihood the population in each generation and determine whether the strategy has converged prematurely or requires a more extensive exploration process function is constructed from the Mean Squared Error (MSE) of the model evaluated under the assumption of the normality of the errors:

$$\text{AIC} = 2k - 2\ln(L). \tag{6}$$

#### 2.4.2 Bayesian Information Criterion

The Bayesian information criterion (BIC) is an essential statistical tool for evaluating and comparing various forecasting methods in the forecasting field [54]. This approach is based on the search for an optimal balance between the forecasting accuracy of the model, evaluated through the likelihood function, and the inherent complexity of the model. A distinctive aspect of the BIC is its ability to penalize those models that benefit from short sample sizes, contributing to a more robust and generalizable evaluation [25]:

$$\text{BIC} = n \cdot \ln(L) + k \cdot \ln(n). \tag{7}$$

In equation 7, $n$ represents the size of the data set, $L$ the likelihood function, and $k$ the number of hyperparameters in the model. Unlike AIC, BIC multiplies the number of parameters by the logarithm of the sample size, a weighting that accentuates the penalty as the complexity of the model increases.

## 3 Proposed Methodology

In this section, we present the design and configurations of HELI (Heuristic Ensemble for Learning Information), a highly flexible and robust methodology for time series forecasting in the context of climate change. The methodology is divided into three fundamental sections: preprocessing, training individual forecast models, and adjusting the participation weighting of each method to form an ensemble.

These phases are shown in detail in Figure 7. Each of these sections houses methods whose configurations are detailed below. These methods are modularly configurable. Thus, removing, extending, or replacing elements with other strategies that perform similar functions is possible.

The methodological structure is detailed in Algorithm 1, which represents the general framework of the methodology. Its input is a time series (with the variable time_series), and its output consists of the optimal weighting obtained together with the evaluation of this weighting using various error metrics.

**Table 1.** Hyperparameter for forecasting models

| Hyperparameter | Value |
|---|---|
| **Random Forest** | |
| Criterion | Friedman mse |
| Estimators | 23000 |
| Maximum features | Sqrt |
| Bootstrap | True |
| Maximum samples | 0.8 |
| Minimum samples leaf | 10 |
| Minimum samples split | 8 |
| **SVR** | |
| Kernel | Poly |
| Degree, Epsilon, C | 5,0.01,5 |
| Gamma | None |
| **Kernel Ridge** | |
| Kernel | Poly |
| Alpha, Gamma | 1, 0.1 |
| Degree, Coef0 | 4, 0.1 |
| **CNN** | |
| **Convolutional Layers** | **2 Conv2D** |
| **Kernel** | **(15,15) & (5,5)** |
| **Filters** | **64** |
| **Strides** | **1** |
| **Pooling** | **Average Pooling** |
| **Dropout** | **0.25** |
| Activation | ReLU |
| Pooling | AverangePooling |
| Optimizer | Adam |
| Learning rate | 0.01 |
| Epochs | 5000 |
| Batch size | 10 |
| **LSTM** | |
| **LTSM Layers** | **1** |
| **LSTM Cells** | **35** |
| **MLP Layers** | **4** |
| **MLP Neurons** | **200** |
| **MLP Dropout** | **0.3** |
| MLP Activation | Tanh |
| Output Activation | Linear |
| Optimizer | Adam |
| Learning rate, Epochs | 0.005,500 |
| Batch size | 10 |
| Early stopping | 40 |

Line 2 describes the data cleaning and processing process (where the function preprocessing cleans the data and is assigned with the variable preprocessing_ts); then, the time series is divided into the three blocks shown in line 3. From lines 4 to 8, individual forecast models are defined, trained, and evaluated using the mean squared error (MSE) metric. This evaluation results in the creation of a heuristic population mentioned in line 9 and detailed in Algorithm 2.

Next, an evolutionary strategy, presented in line 10 and explained in Algorithm 3, explores and exploits the population to find a solution at least as good as the best individual model. This solution reflects the contribution of the individual models to the ensemble, and its evaluation to obtain the resulting error values is indicated in line 11.

The generation of the heuristic population, described in Algorithm 2, takes as input the forecast models together with their corresponding MSE evaluation and returns an initial population of solutions. This population starts empty in line 1, and solutions are generated and subsequently added to it. Delimiting lines 3 to 6, the models are ranked using the BIC and AIC criteria, which rank and score the suitability of the models.

In this context, the models will receive a proportional representation in percentage of the score obtained by the ranking methods, and the generated solutions will be incorporated into the population. For lines 7 and 8, RMSE is used, which is not a ranking method but is used to evaluate the error.

The results of this evaluation make it possible to rank the algorithms from least to most error, generating a percentage inversely proportional to their error and adding the solution to the population. Lines 9 and 10 describe the integration of a uniform solution in which all forecasting models have an equivalent participation in the ensemble.

Lines 11 to 15 detail the integration of solutions where each forecast model receives 100% participation in the ensemble, leaving the others with a participation of 0. This strategy ensures that the results obtained are at least as good as the best individual model.

A random process generates the remaining solutions (lines 16 to 19). These solutions must be feasible in terms of the proportionality of their participation in the ensemble. Algorithm 3 describes an evolutionary strategy to optimize the

**Table 2.** Description of the best configuration found for the evolutionary strategy

| Configuration | Value |
|---|---|
| Number of Generations | Uncertain |
| Population Size | 500 individuals |
| Genotype Size | 7 genes |
| Selection Technique | Weighted Roulette |
| Selected Parents | 200 parents |
| Crossover Type | Scattered |
| Mutation Probability | Adaptative (0.2 – 0.9) |
| Mutation Type | Double Random |
| Allow Duplicates | No |
| Elitism | 50 individuals |
| Stopping Criterion | Rate of change over 20 generations |
| Fitness Objective | Minimize MSE |

weights in the ensemble participation. This algorithm receives the heuristic population, the previously adjusted models, and the validation and training set, returning the best solution found.

In line 2, an evaluation of the population is performed to temporarily store the best solution. From line 4 to line 16, an evolutionary process is run to an evolutionary process is executed that ends when the convergence criterion is met. This process involves the generation of offspring and their mutation, explained in lines 5 and 6. The generated offspring are evaluated using the same metrics and criteria as the initial population, as described in line 9.

When a better solution than the previously stored one is found, it is updated, as shown in lines 10 to 13. Some elements of the population are updated with the generated offspring, as described in line 15. Finally, the algorithm returns the best solution found during the evolutionary process.

The inherent flexibility of HELI translates into the ability to work with various strategies or the incorporation of new ones; however, it is essential to note that such adaptability entails further adjustment and optimization. Nevertheless, the results may vary significantly depending on these factors and the nature of the test data used.

### 3.1 Preprocessing

Given the inherent complexity of the nature of the data, as will be discussed in detail in later sections, a preprocessing of the information is required. This process becomes crucial to ensure the data's integrity and completeness, scale them appropriately, and mitigate the presence of noise and outliers.

Although the methods described find applicability in various time series, it is crucial to recognize that the effectiveness of each strategy may vary according to the specific characteristics of the data. Therefore, the selection of methods must be customized for each data set to be evaluated, adjusting optimally to the particularities of each time series.

An initial review revealed the existence of missing data in the series. However, these were presented in isolation and not consecutively. Opting for data trimming could distort the width of the seasonal spaces, biasing the learning process in specific sections of the series. Therefore, we decided to impute the missing data through quadratic interpolation, thus ensuring the completeness of the series without compromising its temporal structure.

In later stages, we describe two strategies for noise reduction and outlier containment are implemented. Outlier reduction in data sets that may include trend and/or seasonal values represents a challenge since robust techniques applied to other areas such as IQR that base their behavior on median dispersion are not sensitive to seasonal behaviors, so important information may be considered as noise [13].

The 3-sigma technique made it possible to limit the series by eliminating outlier observations. In consideration of the seasonal nature of the series, subject to fluctuations associated with climate change, the 3-sigma strategy uses a moving window whose size represents a seasonal period, allowing relevant information within the seasonal periods to be retained and noise to be suppressed.

The singular Value Decomposition (SVD) process was then applied to the data series. This technique interprets the low-influence components as noise and then allows its smoothing by eliminating them. Subsequently, a singular value decomposition (SVD) process was carried out on

**Table 3.** Statistical information of the data used in the experimentation

| Data filled and re-sampled | | | | | |
|---|---|---|---|---|---|
| | **Count** | **Min** | **Max** | **Media** | **Std Dev.** |
| Min. Temp | 3527 | -2.6° | 14.83° | 8.74° | 2.6546 |
| Max. Temp | 3527 | 7.52° | 32.43° | 22.16° | 2.6836 |

**Table 4**. Average results obtained from the individual tuned forecast models for validation set

| Evaluation Models with sMape | | |
|---|---|---|
| **Models** | **Min Temp** | **Max Temp** |
| SARIMA | 25.42% | 37.40% |
| Holt & Winters | 24.8% | 24.45% |
| CNN | 9.63% | 7.24% |
| LSTM | 11.02% | 6.93% |
| Random Forest | 11.74% | 6.87% |
| SVR | 10.23% | 6.66% |
| Kernel Ridge | 19.96% | 8.41% |

the data series. This technique enables smoothing by eliminating low-influence components that are interpreted as noise.

To address the seasonal nature of the problem, defining the amplitude of the Hankelization matrix as a function of a seasonal period is necessary to preserve 90% of the explained variance components. Climate change studies usually focus on medium and long-term changes, since short-term observations can be affected by other natural or anthropogenic phenomena, leading to high variability [3].

Therefore, weekly or monthly observations provide a better generalization of climate behavior. We resampled the original data collected from daily observations as weekly observations. This process applies an arithmetic mean every seven observations to obtain a weekly data series that facilitates the analysis in a more effective climate forecasting context.

### 3.2 Training Models

The previous section gave an overview of the forecasting methods used in the methodology addressed. The choice of the hyperparameters of the forecasting methods was made through fast tuning using the grid search strategy.

This strategy combines the hyperparameters from a previously defined list, however, to guarantee an optimal performance it is suggested the use of other tuning techniques such as evolutionary algorithms or Bayesian strategies, however, this may significantly impact the model fitting times.

In the same way, the same strategy was used for the tuning of the neural network architecture whose parameters are listed in Table 1.

Table 1, shows the hyperparameters used in each strategy, which were tuned and optimized through genetic algorithms or mesh searches, depending on the intrinsic complexity of each specific algorithm. This process may be performed for each new set of time series, thus ensuring an optimal adaptation to the particular characteristics of each set of information.

Additionally, since the forecasting methods execute and format the data independently, parallel execution was chosen whenever possible, thus optimizing computation time on the processor or GPU. However, this type of execution demands a more considerable amount of working memory. It is important to note that the type of execution, whether parallel, sequential, or mixed, does not affect the results obtained. After training the methods, the Mean Squared Error (MSE) value is obtained for the validation set.

Since this set is invisible in the training process, it provides a more realistic approximation for the models when the entire set of cases is unknown. Next, the construction of the heuristic population begins. We incorporate a weighting process, where the initial set of weights can be determined with the following strategies: a) randomly generated using a uniform distribution, b) an error metric such as RMSE, and c) using ranking metrics, such as AIC or BIC.

Now we describe the ranking generated by BIC and AIC, along with an evaluation of the different models using these metrics. This ranking punctuates the suitability of the models, although they do not explicitly give the percentage of participation of each one.

Taking AIC as an example, any individual weight is given by equation 8, where $AIC_i$

**Table 5.** Average results for different ensembles for validation set

| Ponderations obtained with sMape | | |
|---|---|---|
| **Ponderations** | **Min Temp** | **Max Temp** |
| Uniform | 19.45% | 14.12% |
| AIC | 11.69% | 8.45% |
| BIC | 11.52% | 6.23% |
| **HELI** | 9.12% | 5.87% |

represents the ranking value of each model, $N$ the number of models, and $w_i$ the resulting weight. The weighting process performed in BIC uses a similar equation. The weights process is not necessarily the best, but they give us an initial set of weights taken to explore their neighborhood and evolve toward an optimal set of weights:

$$w_i = \frac{1}{AIC_i \cdot \sum_{j=1}^{N} \frac{1}{AIC_j}}. \quad (8)$$

The initial weights determined for AIC or BIC are taken as solutions in the Ponderation Optimization process described in the next section.

### 3.3 Weighting Optimization

The previous section pointed out that the initial weights obtained by RMSE, AIC, BIC, or a uniform distribution are not necessarily the most adequate. The ranking or selection methods only qualify the most capable strategies but do not determine the degree of participation that each one should have when integrated into the ensemble.

Therefore, it is necessary to adjust their degree of participation to obtain the best possible combination of algorithms. The population in this problem consists of 500 solutions, which are generated by a heuristic strategy, while the rest of the solutions are randomly generated. The order generated by the ranking methods and their obtained values are converted to participation percentages to obtain the heuristic solutions, on the other hand, a solution with uniform weighting is generated, i.e. in which all the algorithms have the same participation in the ensemble.

To ensure that the strategy allows obtaining a solution at least as good as the best individual forecasting method, solutions are also generated where 100% of the ensemble belongs to each of the methods. The parameters of the evolutionary algorithm are detailed in Table 2 likewise, a sub-condition is implemented that is satisfied when the rate of change of the best solutions during ten generations tends to zero.

In this case, a population perturbation is performed on a random amount, logically discarding the elite solutions; this allows us to avoid premature stagnation and explore unexplored sections of the solution space. When the convergence criterion is met, the algorithm returns the best solution found, which will constitute the best weighting of the forecast models and form the ensemble's weights.

The convergence criterion indicates that the algorithm reaches its convergence when the variability of the best solution over 20 generations tends to zero. The value was chosen based on the results of a preliminary test. In this test we observed that improvements of the best solutions occurred at most every 10 or 12 iterations before convergence. That is, when the algorithm stops obtaining improvements.

## 4 Experimentations and Results

This section describes the error metrics used to tune the strategies and generate the results report internally. The specific instances used in the experimental process are presented, along with a complete exposition of all the results obtained.

### 4.1 Experimental Conditions

The proposed methodology and strategies used in the comparison were developed and executed using the Jupyter Notebook interactive environment in Python. Experiments included parallel executions to increase computational efficiency, resulting in significantly faster training time. Specifically, the deep learning neural networks were executed using GPU computing capabilities to minimize learning time and increase their performance.

The equipment used to carry out these tasks has a Ryzen 5700x processor with 32GB of RAM and a Nvidia 4060 graphics processing unit. In all cases, the executions were carried out dynamically

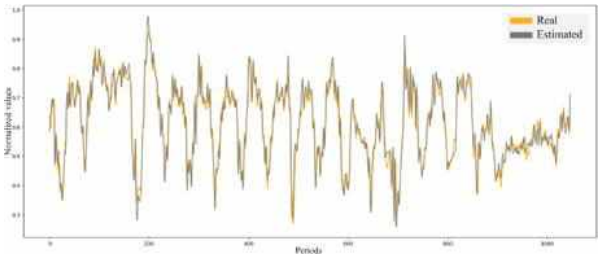



**Fig. 10.** Trend component from 2018 to 2028

with access to hardware resources, with no time constraints imposed. An analysis based on average results derived from at least 30 runs of each strategy is presented to ensure consistency in the predictions and avoid results biased by randomness.

This rigorous replication is applied to the proposed strategies and those from the state of the art, providing a more robust and reliable view of the effectiveness of each approach. Additionally, to ensure the optimality of the results, all models were tuned using specific algorithms.

This tuning phase was carried out comprehensively, adjusting the key parameters of each model to ensure optimal performance in terms of accuracy and generalization. This meticulous approach reinforces the validity and reliability of the results presented in the framework of this comparative analysis.

### 4.2 Error Metrics

Two different types of error metrics are used in the proposed methodology: The Mean Squared Error (MSE) and the Symmetric Mean Absolute Percentage Error (sMAPE). The MSE is used to refine the accuracy of the model and adjust its hyperparameters, while the sMAPE is used exclusively to report model performance.

The MSE is used as an effective tool during the fitting phase, as it amplifies significant errors, thus contributing to the learning process to minimize this error metric. Its formula is expressed in equation 9, where $y_i$ represents the actual observation and $\hat{y}_i$ represents the prediction generated by the model [35]:

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^{N} (y_i - \hat{y}_i)^2. \tag{9}$$

On the other hand, we try to present the results understandably. Thus, we show the sMAPE error in percentage terms, allowing a simple comparison with other methodologies. Its formula is expressed in equation 10, where $y_i$ represents the actual observation and $\hat{y}_i$ represents the prediction generated by the model [35]:

$$sMAPE = \frac{1}{100} \cdot \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{\frac{|y_i|+|\hat{y}_i|}{2}}. \tag{10}$$

The last two metrics provide a comprehensive assessment of the model performance, allowing a detailed understanding of internally adjusted accuracy and externally reported performance.

### 4.3 Data Descriptions

The time series used in this study were derived from meticulously collected observations from meteorological stations, which are part of the Mexican National Meteorological Service (SMN) database [55]. The sample includes data from February 1, 1950, to August 31, 2017, with an hourly observation period.

This reliable source, available for free download at the SMN's official website, is a fundamental resource for analyzing climatic phenomena. The time series, specifically for Maximum and Minimum Temperature, were structured in Celsius degrees, as detailed in Table 3, and reflect carefully recorded observations in the southern region of Mexico City. The data are public, and can be found in the Kaggle repository website [56].

The choice of this geographical area was not arbitrary; on the contrary, it is based on carefully selected criteria. This locality encompasses the areas of Milpa Alta, Xochimilco, and Tlahuac, regions with specific climatic relevance that are observable in Figure 8. The vulnerability of the Valley of Mexico to climate change is evidenced by the significant population density of this region, the most populated in the country.

The marked anthropogenic influence derived from industrialization and the consequent generation of pollutants makes this area a crucial enclave for understanding the effects of climate change in an urban context [57]. Additionally, the choice of the southern zone of Mexico City responds to its particular orographic condition. This

**Table 6.** Average results were obtained from HELI and compared to the best state-of-the-art strategy for test set

| Evaluation Methodologies with sMape | | |
|---|---|---|
| Methods | Min Temp | Max Temp |
| SMYL | 8.78% | 5.13% |
| HELI | 9.54% | 6.02% |

**Table 7.** Wilcoxon test shows Heli and SMYL are equivalent

| Statistical results | |
|---|---|
| ∞ | 0.05 |
| Z-statistic | 2.126 |
| p-value | 0.0554 |

geographical feature mitigates the heat island effect, a common phenomenon in urbanized areas.

The exclusion of this effect allows a more accurate analysis of climate variations and the impacts of climate change in the region. The methodology included to subdividing the data into three distinct blocks: the training set, validation, and test, allocating 60% for training and 20% for each remaining set, respectively.

The training block played a key role in the process, as it was used to train each forecast model and adjust the ensemble weights. This methodological approach contributes to a comprehensive and robust evaluation of forecast models in forecasting.

### 4.4 Results Obtained

The first experimental phase focuses on the adjustment and optimization of the individual forecasting methods. Table 4 presents the results obtained for each of them in both instances, expressed in terms of the scaled mean absolute mean percentage error (sMAPE) for the validation set, where an ensemble with which the models are not trained is evaluated.

The effectiveness of forecasting methods depends exclusively on the forecasting model, its hyperparameter configuration and the data. In this case CNN and SVR present the best results in general, however for other instances it is not guaranteed that they have similar effectiveness, the “No free lunch theorem” suggests that there are no universally superior methods for all problems raised in machine learning [14].

Machine learning models outperform classical forecasting models, attributable to the complexity of the data and the presence of natural temperature changes treated as noise. CNN, LSTM, and Random Forest generally exhibit superior performance.

However, SVR and Kernel Ridge show good results with fewer adjustable parameters, which could mitigate the risk of overfitting. Besides, the weights of metrics are adjusted by a heuristic population that evolves to find an optimal combination, thus issuing quality forecasts. Table 4 shows that classical methods have a more limited performance than machine learning methods due to the better generalization of the latter for time series complexity.

Neural and Random Forest strategies stand out, although they require more hyperparameter tuning and training time. In Table 5 includes the performance of models with ensemble solutions. The uniform ensemble worsens most of the individual models since it does not consider the differences in the generalization of the curves.

Although the results improve with metrics such as BIC and AIC, they are not guaranteed to outperform the best individual method. The optimized forecast outperforms the individual methods and the evaluated ensemble strategies thanks to an initial population that includes other ensemble weights.

In addition, the proposed strategy is compared with state-of-the-art ensemble approaches in Table 6, such as SMYL. The latter was built from the winning paper of the M4 prognostics competition, with similar modifications to the preprocessing of the proposed strategy and hyperparameter tuning by genetic algorithm. Although SMYL shows a slight superiority in the tests shown in Table 7, a Wilcoxon test reveals no significant differences, indicating equivalence between both strategies.

In terms of time, the average time for HELI is barely 20 minutes per execution, while SMYL takes 1 minute if it already knows the instance and has a semi-empty pool, and 25 minutes if it does not.

This value can be misleading since the Smyl strategy is designed to have a pool of solutions, so


**Fig. 9.** HELI results compared to real data

**Table 8.** Estimated values of the proposed model compared to the actual values from 2018 onwards

| Forecasting results | | | | |
|---|---|---|---|---|
| | Minimum Temp | | Maximum Temp | |
| Year | Real | Estimated | Real | Estimated |
| 2018 | 10.2° | 10.5° | 23.5° | 22.7° |
| 2019 | 10° | 10.6° | 23.3° | 22.9° |
| 2020 | 10.1° | 10.4° | 23.2° | 22.7° |
| 2021 | 9.8° | 10.3° | 23° | 22.8° |
| 2022 | 9.9° | 10.4° | 23.1° | 22.8° |
| 2023 | 10° | 10.2° | 23.2° | 23° |

**Table 9.** Results for second experiment

| sMAPE forecasting evaluation | | |
|---|---|---|
| Variables | Monterrey | Guadalajara |
| Max Temp | 10.8% | 9.5% |
| Min Temp | 12.7% | 11.5% |
| $CO_2$ | 6.4% | 5.6% |
| $NO_2$ | 4.3% | 6.4% |
| $PM_{10}$ | 7.2% | 5.1% |

the more knowledge it has, the longer it will take to execute [19].

### 4.5 Interpretation of the Forecasts Obtained

According to the regression curve obtained, the actual results can be contrasted with those obtained through the proposed forecasting method. This comparison is illustrated in Figure 9, where the inherent complexity of the forecasting task is evident, as well as the ability of the ensemble approach to integrate strategies, effectively capturing trend, seasonality, and even random noise patterns.

### 4.6 Results Discussion

This study focuses on presenting the results using the sMAPE metric; however, it is crucial to highlight that other metrics allow for discerning the effectiveness of the proposed strategy compared with the leading method in state-of-the-art.

In particular, the importance of considering training time in this evaluation is emphasized. Smyl was executed under the same experimental conditions as HELI; however, the architecture proposed by SMYL does not implement parallel structures, so its execution is sequential.

The proposed strategy, HELI, and SMYL exhibit similar training times in the analyzed instances. However, it is essential to note that SMYL accumulates knowledge as it is tested with various time series.

In this context, the raw learning time taken by Smyl is considerably larger to achieve its results. In contrast, the HELI method presents a constant adjustment time, depending only on the individual complexity of the series and its size.

From this perspective, HELI provides equivalent results in a lower adjustment time; thus, it offers a practical and time-efficient alternative. The forecasting tasks were not limited to the initial test set.

### 4.7 Prediction Over New Horizons

We performed an additional test to evaluate the ability of HELI to predict a longer period, by extending the forecast beyond the limits of the actual time series data until 2028. For this experiment we expanded the forecast period in 521 weeks in the future, which is equivalent to 10 years ahead. In other words, we projected the prediction of HELI since 2018 to 2028. Although the data set is limited to 2018, at the time of writing this paper is possible to contrast the results up to 2023. In Table 8, we can observe a clearly convergence between the estimated annual averages and the actual annual values. These results reveal the effectiveness of HELI.

In addition, Figure 10 shows the estimated evolution of the trend component, which presents a slight increase in the gap between maximum and minimum temperatures as we move into the future. This finding suggests that long-term

weather patterns in southern Mexico City are undergoing changes, which could indicate an influence of phenomena associated with climate change.

### 4.8 Evaluation for Other Climate Change Instances

To evaluate the effectiveness of the method, we analyzed other instances of climate change-related variables, this time for the Mexican cities of Guadalajara and Monterrey. We used public data collected from airport stations.

Table 9, shows the results obtained for the variables of maximum temperature, minimum temperature, the gases $CO_2$, and $NO_2$ and the $PM_{10}$ particles. We performed this experiment with 4012 observations, and we divided the data in the same way as in the first experiment. In the second experiment, we adjusted the hyperparameters of the forecasting algorithms.

The sMAPE results show that the accuracy is maintained, with error values ranging since 4.3 to 12.7% for Monterrey and 5.1 to 11.5% for Guadalajara. Notice that in this second experiment, Heli obtained the best forecast for the two gases, followed by PM10 particles. Besides, the forecast for the two temperature variables is reasonably acceptable.

## 5 Conclusions and Discussions

Climate change represents one of the most decisive challenges for the present and future of humanity. Its profound environmental, economic, and social implications demand the urgent development of effective methods for temperature forecasting. In this context, this paper presented HELI, a novel methodology for temperature forecasting that integrates robust preprocessing, training of several individual methods, and optimization of their weights through evolutionary algorithms. Experiments performed on extensive temperature time series in the southern area of Mexico City show promising results.

HELI achieves a sMAPE error of 9.54% for minimum and 6.02% for maximum temperature, outperforming both state-of-the-art individual methods and sophisticated ensemble strategies previously in the state of the art. Rigorous statistical tests demonstrate HELI's equivalence to SMYL, the leading global method. HELI's competitiveness is even more valuable when considering its flexibility and scalability.

Unlike other approaches, HELI does not accumulate knowledge between time series but performs a specific adjustment for each one. This translates into bounded and constant training times, dependent only on the individual complexity of each series. On the other hand, HELI's modularity allows easy incorporation of new methods and configurations. The solid results obtained position HELI as a highly effective alternative for temperature prediction.

This work lays the groundwork for further exploration of integrations with new individual methods, improved preprocessing, and more comprehensive hyperparameter optimizations. It remains a pending task to extend the evaluation of HELI to other regions and types of time series related to climate phenomena. In conclusion, HELI represents a significant advance in the crucial and urgent field of climate forecasting.

Both its methodological design and experimental contributions are concrete contributions to develop increasingly effective solutions to the complex and urgent challenges posed by climate change.

HELI's potential for flexibility, enabling the assessment of multiple individual variables; scalability, allowing the incorporation of other forecasting variables, and competitiveness lay a solid foundation for its potential adoption in climate studies. We hope this work will stimulate new lines of research in this priority area for the planet's sustainability.

## Acknowledgments

The authors would like to acknowledge with appreciation and gratitude to CONAHCYT, TecNM, and Laboratorio Nacional de Tecnologías de la Información(LaNTI) in the Instituto Tecnológico de Ciudad Madero for the access to the cluster. Erick Estrada Patiño would like to thank CONAHCYT for the Ph.D. Scholarship.

## References

1. **Raffel, T. R., Romansic, J. M., Halstead, N. T., McMahon, T. A., Venesky, M. D., Rohr, J. R. (2012).** Disease and thermal acclimation in a more variable and unpredictable climate. Nature Climate Change, Vol. 3, No. 2, pp. 146–151. DOI: 10.1038/nclimate1659.
2. **Moss, R. H., Edmonds, J. A., Hibbard, K. A., Manning, M. R., Rose, S. K., van-Vuuren, D. P., Carter, T. R., Emori, S., Kainuma, M., Kram, T., Meehl, G. A., Mitchell, J. F. B., Nakicenovic, N., Riahi, K., Smith, S. J., Stouffer, R. J., Thomson, A. M., Weyant, J. P., Wilbanks, T. J. (2010).** The next generation of scenarios for climate change research and assessment. Nature, Vol. 463, No. 7282, pp. 747–756. DOI: 10.1038/nature08823.
3. **Masini, R. P., Medeiros, M. C., Mendes, E. F. (2021).** Machine learning advances for time series forecasting. Journal of Economic Surveys, Vol. 37, No. 1, pp. 76–111. DOI: 10.1111/joes.12429.
4. **Al-Ghussain, L. (2018).** Global warming: Review on driving forces and mitigation. Environmental Progress and Sustainable Energy, Vol. 38, No. 1, pp. 13–21. DOI: 10.1002/ep.13041.
5. **Hardy, J. T. (2003).** Climate change: Causes, effects, and solutions. John Wiley and Sons.
6. **Houghton, J. T., Jenkins, G. J., Ephraums, J. J. (1990).** Climate change: The IPCC scientific assessment. American Scientist (United States), Vol. 80, No. 6.
7. **Peters-Lidard, C. D., Rose, K. C., Kiang, J. E., Strobel, M. L., Anderson, M. L., Byrd, A. R., Kolian, M. J., Brekke, L. D., Arndt, D. S. (2021).** Indicators of climate change impacts on the water cycle and water management. Climatic Change, Vol. 165, No. 1–2. DOI: 10.1007/s10584-021-03057-5.
8. **Hou, J., Wang, Y., Zhou, J., Tian, Q. (2022).** Prediction of hourly air temperature based on CNN–LSTM. Geomatics, Natural Hazards and Risk, Vol. 13, No. 1, pp. 1962–1986. DOI: 10.1080/19475705.2022.2102942.
9. **Ray, S., Das, S. S., Mishra, P., Al-Khatib, A. M. G. (2021).** Time series SARIMA modelling and forecasting of monthly rainfall and temperature in the south Asian countries. Earth Systems and Environment, Vol. 5, No. 3, pp. 531–546. DOI: 10.1007/s41748-021-00205-w.
10. **Pang, B., Yue, J., Zhao, G., Xu, Z. (2017).** Statistical downscaling of temperature with the random forest model. Advances in Meteorology, Vol. 2017, pp. 1–11. DOI: 10.1155/2017/7265178.
11. **Papacharalampous, G., Tyralis, H., Koutsoyiannis, D. (2018).** Predictability of monthly temperature and precipitation using automatic time series forecasting methods. Acta Geophysica, Vol. 66, No. 4, pp. 807–831. DOI: 10.1007/s11600-018-0120-7.
12. **Radhika, Y., Shashi, M. (2009).** Atmospheric temperature prediction using support vector machines. International Journal of Computer Theory and Engineering, Vol. 1, No. 1, pp. 55–58. DOI: 10.7763/ijcte.2009.v1.9.
13. **Wolpert, D., Macready, W. (1997).** No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, Vol. 1, No. 1, pp. 67–82. DOI: 10.1109/4235.585893.
14. **Dietterich, T. G. (2000).** Ensemble methods in machine learning. International Workshop on Multiple Classifier Systems, pp. 1–15. DOI: 10.1007/3-540-45014-9_1.
15. **Frausto-Solis, J., Rodriguez-Moya, L., González-Barbosa, J., Castilla-Valdez, G., Ponce-Flores, M. (2022).** FCTA: A forecasting combined methodology with a threshold accepting approach. Mathematical Problems in Engineering, Vol. 2022, No. 1, pp. 1–19. DOI: 10.1155/2022/6206037.
16. **Yan, G., Yu, C., Bai, Y. (2021).** A new hybrid ensemble deep learning model for train axle temperature short term forecasting. Machines, Vol. 9, No. 12, pp. 312. DOI: 10.3390/machines9120312.
17. **Makridakis, S., Spiliotis, E., Assimakopoulos, V. (2018).** The M4 competition: results, findings, conclusion and way forward. International Journal of

Forecasting, Vol. 34, No. 4, pp. 802–808. DOI: 10.1016/j.ijforecast.2018.06.001.

18. **Smyl, S. (2020).** A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. International Journal of Forecasting, Vol. 36, No. 1, pp. 75–85. DOI: 10.1016/j.ijforecast. 2019.03.017.
19. **Gelper, S., Fried, R., Croux, C. (2009).** Robust forecasting with exponential and holt–winters smoothing. Journal of Forecasting, Vol. 29, No. 3, pp. 285–300. DOI: 10.1002/ for.1125.
20. **Alpaydin, E. (2020).** Introduction to machine learning, fourth edition. MIT Press.
21. **Ahmed, N. K., Atiya, A. F., Gayar, N. E., El-Shishiny, H.** (**2010).** An empirical comparison of machine learning models for time series forecasting. Econometric Reviews, Vol. 29, No. 5–6, pp. 594–621. DOI: 10.1080/0747 4938.2010.481556.
22. **Engelhardt, I., De-Aguinaga, J., Mikat, H., Schüth, C., Liedl, R. (2013).** Complexity vs. simplicity: groundwater model ranking using information criteria. Groundwater, Vol. 52, No. 4, pp. 573–583. DOI: 10.1111/gwat.12080.
23. **Chakrabarti, A., Ghosh, J. K. (2011).** AIC, BIC and recent advances in model selection. Philosophy of Statistics, Vol. 7, pp. 583–605. DOI: 10.1016/b978-0-444-51862-0.50018-6.
24. **Kuha, J. (2004).** AIC and BIC: comparisons of assumptions and performance. Sociological Methods and Research, Vol. 33, No. 2, pp. 188–229. DOI: 10.1177/0049124103262065.
25. **Faloutsos, C., Gasthaus, J., Januschowski, T., Wang, Y. (2018).** Forecasting big time series: old and new. Proceedings of the VLDB Endowment, Vol. 11, No. 12, pp. 2102–2105. DOI: 10.14778/3229863.3229878.
26. **Peter, G., Matskevichus, M. (2019).** Hyperparameters tuning for machine learning models for time series forecasting. 6th International Conference on Social Networks Analysis, Management and Security, Vol. 6, pp. 328–332. DOI: 10.1109/snams.2019.893 1860.
27. **Makridakis, S., Hibon, M. (1997).** ARMA models and the box-Jenkins methodology. Journal of Forecasting, Vol. 16, No. 3, pp. 147–63. DOI: 10.1002/(sici)1099-131x(199705)16; 3<147::aid-for652>3.0.co;2-x.
28. **Bustos, O. H., Yohai, V. J. (1986).** Robust estimates for ARMA models. Journal of the American Statistical Association, Vol. 81, No. 393, pp. 155–168. DOI: 10.1080/01 621459.1986.10478253.
29. **Fang, T., Lahdelma, R. (2016).** Evaluation of a multiple linear regression model and SARIMA model in forecasting heat demand for district heating system. Applied Energy, Vol. 179, pp. 544–552. DOI: 10.1016/j.apenergy.20 16.06.133.
30. **Hyndman, R. J., Athanasopoulos, G. (2018).** Forecasting: Principles and practice. OTexts.
31. **Chatfield, C. (1978).** The holt-winters forecasting procedure. Applied Statistics, Vol. 27, No. 3, pp. 264–279. DOI: 10.2307/23 47162.
32. **Indriani, R., Sugandha, A., Tripena, A., Larasati, N., Rokhman, A. F., Bon, A. T. (2020).** Forecasting of air temperature in Cilacap Regency with triple exponential smoothing (holt-winter) method. 5th NA International Conference on Industrial Engineering and Operations Management.
33. **Ali, J., Khan, R., Ahmad, N., Maqsood, I. (2012).** Random forests and decision trees. International Journal of Computer Science Issues, Vol. 9, No. 5, p. 272
34. **Breiman, L. (2001).** Random forests. Machine learning, Springer, Vol. 45, No. 1, pp. 5–32. DOI: 10.1023/a:1010933404324.
35. **Esmaily, H., Tayefi, M., Doosti, H., Ghayour-Mobarhan, M., Nezami, H., Amirabadizadeh, A. (2018).** A comparison between decision tree and random forest in determining the risk factors associated with type 2 diabetes. Journal of Research in Health Sciences, Vol. 18, No. 2, pp. 412.
36. **Altman, N., Krzywinski, M. (2017).** Ensemble methods: Bagging and random forests. Nature Methods, Vol. 14. No. 10, pp. 933–935.
37. **Ziegler, A., König, I. R. (2013).** Mining data with random forests: Current options for real-world applications. WIREs Data Mining and

Knowledge Discovery, Vol. 4, No. 1, pp. 55–63. DOI: 10.1002/widm.1114.

38. **Sapankevych, N., Sankar, R. (2009).** Time series prediction using support vector machines: a survey. IEEE Computational Intelligence Magazine, Vol. 4, No. 2, pp. 24–38. DOI: 10.1109/mci.2009.932254.
39. **Cortes, C., Vapnik, V. (1995).** Support-vector networks. Machine Learning, Vol. 20, No. 3, pp. 273–297. DOI: 10.1007/bf00994018.
40. **Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T. (2018).** Recent advances in convolutional neural networks. Pattern Recognition, Vol. 77, pp. 354–377. DOI: 10.1016/j.patcog.2017.10.013.
41. **O'Shea, K., Nash, R. (2015).** An Introduction to Convolutional Neural Networks. DOI: 10.48 550/arXiv.1511.08458.
42. **Galajit, K., Karnjana, J., Unoki, M., Aimmanee, P. (2019).** Semi-fragile speech watermarking based on singular-spectrum analysis with CNN-based parameter estimation for tampering detection. APSIPA Transactions on Signal and Information Processing, Vol. 8, No. 1. DOI: 10.1017/atsip. 2019.4.
43. **Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., Schmidhuber, J. (2017).** LSTM: A search space odyssey. IEEE Transactions on Neural Networks and Learning Systems, Vol. 28, No. 10, pp. 2222–2232. DOI: 10.1109/tnnls.2016.2582924.
44. **Hochreiter, S., Schmidhuber, J. (1997).** Long short-term memory. Neural Computation, Vol. 9, No. 8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
45. **Gers, F. A., Schmidhuber, J., Cummins, F. (2000).** Learning to forget: continual prediction with LSTM. Neural Computation, Vol. 12, No. 10, pp. 2451–2471. DOI: 10.1162/089976600 300015015.
46. **Taillardat, M. (2017).** Non-parametric Methods of post-processing for Rnsemble Forecasting. PhD Thesis, Université Paris Saclay. theses.hal.science/tel-01723573/.
47. **Estrada-Patiño, E., Castilla-Valdez, G., Frausto-Solís, J., Terán-Villanueva, J. D. (2017).** Enfoque de aprendizaje híbrido evolutivo para redes neuronales en la clasificación de casos médicos. Programación Matemática y Software, Vol. 9, No. 3. DOI: 10. 30973/progmat/2017.9.3/8.
48. **Holland, J. H. (1992).** Genetic algorithms. Scientific American, Vol. 267, No. 1, pp. 66– 73.
49. **Bozdogan, H. (1987).** Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. Psychometrika, Vol. 52, No. 3, pp. 345–370. DOI: 10.1007/bf02294361.
50. **Yamaoka, K., Nakagawa, T., Uno, T. (1978).** Application of Akaike's information criterion (AIC) in the evaluation of linear pharmacokinetic equations. Journal of Pharmacokinetics and Biopharmaceutics, Vol. 6, No. 2, pp. 165–175. DOI: 10.1007/bf01 117450.
51. **Neath, A. A., Cavanaugh, J. E. (2011).** The bayesian information criterion: background, derivation, and applications. WIREs Computational Statistics, Vol. 4, No. 2, pp. 199–203. DOI: 10.1002/wics.199.
52. **Gobierno de México (2022).** Servicio Meteorológico Nacional. smn.conagua. gob.mx/es/
53. **Gobierno de México (2024).** Servicio Meteorológico Nacional: Máximas y mínimas de temperatura–Ciudad de México, México. www.kaggle.com/datasets/erickestradapatio/ maximum-and-minimum-temperatures-cdmx-mxico.
54. **Instituto Nacional de Estadística y Geografía (2022).** Mapas. Climatológicos. www.inegi.org.mx/temas/climatologia/.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Juan Frausto Solis.*

ISSN 2007-9737

# Harmonization of Knowledge Representation: Integrating Systems Thinking Ideas with Appropriate Domain Representation Strategies

Karla M. Olmos-Sánchez[1,*], Jorge Rodas-Osollo[1], Aidé Aracely Maldonado-Macías[2], Alicia Jiménez-Galina[1]

[1] Universidad Autónoma de Ciudad Juárez, Departamento de Ingeniería Eléctrica y Computación, Mexico

[2] Universidad Autónoma de Ciudad Juárez, Departamento de Ingeniería Industrial y Manufactura, Mexico

{kolmos, jorge. rodas, amaldona}@uacj.mx, al206560@alumnos.uacj.mx

**Abstract.** The cognitive era is marked by the prevalence of Artificial Intelligence and information technologies, which implies digital transformation. However, the swift changes and oversimplified synthesis applied to complex domains have resulted in inadequate transformations, particularly in addressing intricate contexts. This paper examines the need for a methodological framework to guide the conceptualization and development of smart solutions. Moreover, the paper underscores the need to establish a comprehensive framework combining Knowledge Management and Systems Thinking to tackle complex situations effectively. It accentuates the significance of comprehending diverse perspectives, exploring solutions alternatives, and managing tacit knowledge within rapidly evolving domains. The integration of Knowledge Management and Systems Thinking through the KMoS-SSA framework allows the difficulty of complex domains to be effectively addressed, facilitating the conceptualization and development of intelligent solutions that are feasible, effective and desirable.

**Keywords.** KMoS-SSA, systems thinking, soft system methodology, complex domains, cognitive era, tacit knowledge, knowledge management.

## 1 Introduction

In a world where technology plays a fundamental role, we are immersed in a significant digital transformation. In this evolution, digital technologies leverage data to drive intelligent workflows, achieving more agile decision-making and real-time responses to environmental disturbances [1]. Those who have already carried out this transformation are entering the beginning of the Cognitive Era, where Artificial Intelligence (AI) and Information Technologies allow them to obtain, assimilate and adapt data, information, and knowledge, which also facilitates decision making and the generation of desired behaviors [2].

The Cognitive Era, considered the current phase of humanity's technological evolution, has created an anxiety to take advantage of all its tools generating hasty changes, which have also caused inadequate digital or cognitive transformations [3].

This is mainly due to an exaggerated synthesis that oversimplifies inherent complex domains, with the intention of keeping them affordable and ensuring the success of AI tools. Complex domains refer to fuzzy-defined areas that present a high number of variables, interactions, or elements, where the relationships between these elements are usually non-linear and may involve many interdependent factors.

They are characterized by their dynamism and the presence of emerging behaviors, which makes it difficult to treat them correctly. The excessive simplification in this kind of domains, not only limits support for strategic decision-making, but also hinder the necessary changes required in the operations of all actors within the domain.

Given this complexity, there is a significant expectation for innovative actions. Nevertheless, it

is crucial to confront these challenges while preserving the unique features of complex domains.

This entails developing alternatives that consider these features and facilitate feasible, effective, and desirable solutions. A feasible solution is one that can be implemented within the constraints of the situation, considering factors such as resources, time, and practicality. An effective solution not only addresses the problem at hand but also produces the desired outcomes.

Finally, a desirable solution goes a step further, being not only effective but also preferred or considered beneficial in terms of satisfaction, convenience, or value. Otherwise, an inadequate understanding of such domains could result in the creation of negligent and insufficient solutions–quasi-solutions–that fail to support effective decision-making.

Specially, this paper describes the methodological framework KMoS-SSA [4], which is designed to guide the conceptualization, specification, and development of intelligent solutions in complex domains using Knowledge Management (KM) and Systemic Thinking (ST). This systemic approach should not only consider the diverse perspectives and alternative solutions of actors but also how they use and share information and knowledge.

Also, these solutions must be effective and align with client needs, particularly in contexts where multiple interconnected elements, ambiguity, and uncertainty are prevalent, leading to emergent properties. In such domains, the knowledge of the highly specialized professionals involved evolves rapidly, resulting in different and sometimes contradictory worldviews among these Domain Specialists (DS).

Tacit Knowledge (TK) becomes paramount, necessitating a systemic vision and effective TK management. Our approach diverges from traditional AI development, which often oversimplifies domains and struggles when addressing the complexities of identifying and conceptualizing solutions in complex environments. ST and related methodologies, such as Soft Systems Methodology (SSM), Strategic Options Development and Analysis (SODA) and Cognitive Mapping Technique, have demonstrated efficacy in proposing solutions for various complex domains.

For instance, within the educational sphere, numerous intricate issues arise, such as enhancing university promotion strategies [5]. Similarly, challenges manifest in diverse areas like the management of hazardous medical waste [6], the design of interactive artifacts in a blockchain-based precision healthcare ecosystem [7], and decision support within the coffee agro-industry [8].

Employing a systemic approach, specifically SSM, has facilitated the decomposition of these complex scenarios into fundamental components. This process reveals areas of conflict and discrepancy among different actors and stakeholders in each context.

However, as the demand for artificial intelligence in addressing organizational needs increases, we propose that integrating knowledge and systemic management into a methodological framework could facilitate the continuous adaptation of knowledge in response to evolving circumstances.

This approach is envisioned to help tackle the challenges posed by complex situations. The proposed methodological framework draws from Knowledge Management (KM), specifically KMoS-RE, and systems thinking, particularly Soft Systems Methodology (SSM), as detailed in subsequent sections. Additionally, we aim to expound upon the underlying theoretical and methodological foundations that serve as the framework's underpinning.

The structure of the paper is as follow: Section 2 will provide background information on Complex Domains, Systemic Thinking and Knowledge Management. Section 3 will describe the methodological framework KMoS-RE-SSA tailored to address the unique characteristics of this kind of domains. In section 4, an analysis of the integration of Systemic Thinking and Knowledge Management will be analyzed.

We will explore how these two perspectives can be effectively combined to enhance understanding and decision-making in complex domains. Finally, we will conclude with a summary of our proposals and suggestions for future research in this area.

## 2 Background

### 2.1 Complex Domains

In the present Cognitive Era, models that communicate perspectives of the real world are integrated into cognitive ecosystems [2].

These ecosystems define domains where decisions are made, and actions are taken to address problematic situations [3]. To address solutions to such problems, human actors perform high-level mental functions to make decisions based on heuristics, unconscious rules of thumb, emotions, clumsiness, insights, and shortcuts; that is, tacit knowledge. Unfortunately, with the rise of computer technology in the last century, there has been a tendency to oversimplify models and analyses of various domains.

This oversimplification often excludes the incorporation of human thinking, intelligence, intuition, and experiential knowledge into the conceptualization and specification of solutions. These aspects, rooted in individual and collective human situations, are shaped by cultural, idiosyncratic, and societal factors, including customs and practices [9].

The progression of AI and the unfulfilled quest for a general AI have prompted continued reflection on how to interact with the real world, that means cognitive ecosystems, and the domains within them. Consequently, depending on the problems we intend to address or solve, especially those that involve a significant human component, the way to address them must be appropriately selected, which implies understanding what the inherent characteristics of complex domains are.

A notable example illustrating these domains is work-related stress. Stress is the reaction to one or more psychosocial risks that can impact a person's mental or physical health and well-being. Therefore, when those risks are found in work organization, work design, and labor relations, it is called work-related stress.

It occurs when the demands of the job do not match or exceed the capabilities, resources, or needs of the worker or when the knowledge or abilities of an individual worker or group to cope are not matched with the expectations of the organizational culture of an enterprise or organization [10].

New modalities of work have imposed new challenges to study these risks and the knowledge representation necessary to prevent and diminish them in these contexts. This complex problem is further exacerbated within the Cognitive Era and holds substantial significance due to its adverse implications for public health, economy, organizations, and society at large.

Work-related stress is inherently complex due to several factors:

- The diversity of the workforce, encompassing different hierarchical levels and responsibilities, introduces a myriad of perspectives and reactions to stressors within the organizational dynamics.
- A multitude of actors, including employees, supervisors, and specialists such as doctors and psychologists, contribute to the intricacy of the situation.
- The existence of various theoretical models attempting to explain the phenomenon from diverse angles adds to its complexity. Examples include the Demand-Control Model [11], Effort-Reward Imbalance Model [12], Social Support Theory [13], Stress Burnout Theory [14], Transactional Stress Model Theory [15], Analysis of Mental Fatigue [16] among others.
- Organizational dynamics, unique to each context, necessitate tailor-made solutions.
- Government regulations introduce an additional layer of complexity.

#### 2.1.1 Complex Informal Structured Domains

Currently, conceptualizing and developing AI solutions requires a transdisciplinary work group with highly specialized knowledge, due to its complexity. This team is generally foreign, and therefore neophyte, to the domain in which the solution will be implemented. When this situation occurs, a Complex Informal Structured Domain (CISD) is generated.

CISD are an evolution of Informal Structured Domains (ISD). In [3], a model addressing ISD with pertinent characteristics, encompassing even

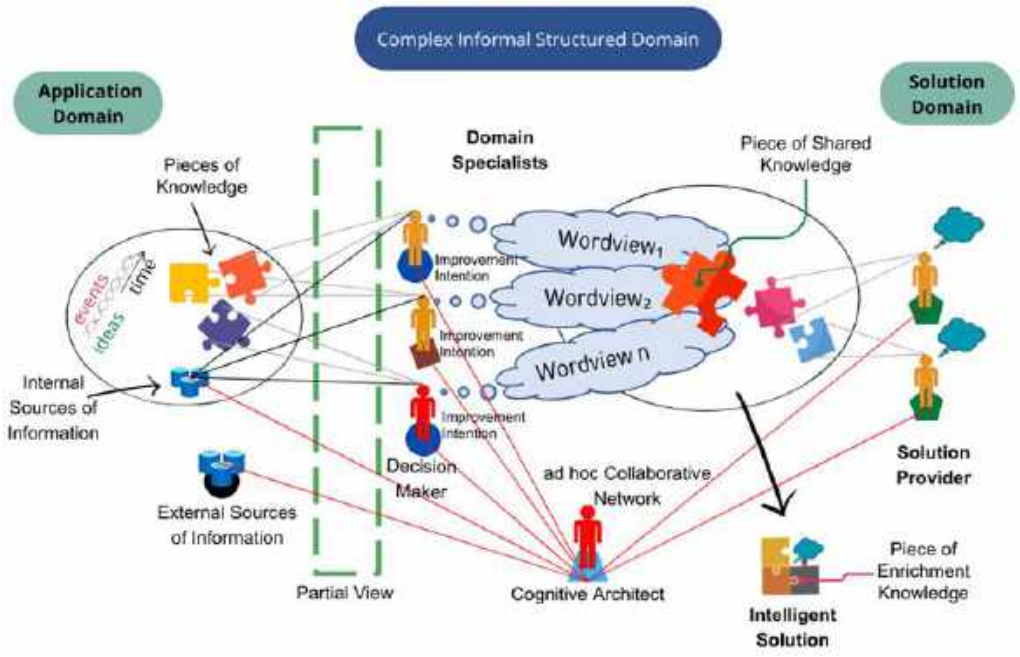



**Fig. 1.** Complex informally structured domain (own creation)

Complex Informal Structured Domains (CISD), is delineated. The description of this complex domain can be formulated as follows:

- Addressing the challenges inherent in ISD demands the expertise of a specialized team of professionals.
- The data, information, and knowledge within ISD exhibit heterogeneity.
- Most of the knowledge in ISD is tacit, lacking structure, and impractical to communicate effectively.
- Furthermore, highly specialized knowledge, whether partially explicit or even explicit, tends to be informal and characterized by a deficient information structure.
- Overall, ISD is marked by high levels of ambiguity and uncertainty.

CISD is distinguished by the presence of its actors, encompassing their cognitive processes, behaviors, and interactions with entities. The components within this domain demonstrate intricate interconnections addressing diverse levels of knowledge and experience, where boundaries are inherently fuzzy. Consequently, collaborative work unfolds in a social, cultural, intuitive, and consensual manner. The actors share interconnections containing both well-structured explicit information and unstructured or loosely structured knowledge.

These elements collectively contribute to comprehending the nature of the problem or need. Consequently, multiple perspectives of the phenomenon defining the complex domain emerge, yielding one or more alternatives for addressing it, with or without an algorithmic solution. The visual representation of the CISD is depicted in Fig. 1. The Application Domain (problematic situation area) portrays the domain in which Domain Specialists (DS) actively participate.

These specialists can come from different disciplines, can have partial knowledge depending on their role in the domain, and may assume one or more roles.

The knowledge possessed by these DS can be explicit (represented by the puzzle´s pieces), but is predominantly tacit, non-homogeneous, informal, and unstructured. Concepts and relationships within this domain are characterized by its ambiguity and are derived from the context and experience of the DS.

Domain behavior exhibits dynamism and emergence, with a complex and interactive flow of events and ideas that evolve over time, influenced by both internal and external sources of information. Additionally, DS hold a partial view of the application domain and diverse worldviews regarding problem situations (represented by clouds), despite sharing knowledge.

It is expected that each DS is committed to enhancing the problem situation by having an improvement intention. The Solution Domain comprises a group of Solution Providers (SP), with the Cognitive Architect leading the providers and overseeing negotiation processes with DS through the ad hoc collaborative network.

Solution Providers also have tacit and explicit knowledge, contributing to the development of effective, feasible and desirable satisfaction resources through a set of artefacts that together form the domain's cognitive architecture. The satisfaction resource may manifest as a tangible or intangible intelligent solution containing enriched knowledge.

### 2.1.2 Exploring the Implications of CISD

Work stress illness is a physical and mental health problem that affects 75% of workers in Mexico. The causes of work stress in this country are diverse and can include work overload, workplace harassment, lack of support from superiors, lack of autonomy at work, among others.

To address this health problem, national authorities through the Ministry of Labor and Social Welfare (STPS) have developed regulations and programs such as NOM-035-STPS-2018 [17,18]. Consequently, organizations need to implement strategic solutions to solve current regulations and improve the productivity and performance of workers.

Conceptualizing, designing, and implementing intelligent solutions to address this phenomenon in organizations is presented as a CISD, due to:

- The phenomenon of work stress is a complex domain (section 2.1).
- The presence of multiple domain specialists, such as managers, occupational physicians, workers, decision makers, etc.
- The presence of a group of solution providers specialized in AI techniques, information technologies, development methodologies, etc.
- Diversity of scientific theories of work stress.
- Internal information in organizations related to work stress (Internal source of information).
- Regulations that influence the way the phenomenon is approached in an organization (External sources of information).
- Different worldview of domain specialists, which can converge in certain aspects, and which must be considered to conceptualize solution alternatives.

Recognizing that the challenge of assimilation in CISD is intricate and necessitates consideration of numerous factors is imperative. Typically, individuals experiencing a situation, problem, or need look for immediate solutions without being fully aware of the situation. This lack of awareness stems from the dynamic and ever-changing nature of activities related to the application domain, rendering prevention unfeasible. While the organization and processes may function acceptably under standard conditions, survival in the Cognitive Era mandates conscious innovation.

Facilitating this innovation necessitates alterations, interactions, and interrelationships among processes, their actors, and the communication channels between them. Domain knowledge remains uncertain and ambiguous, predominantly residing with a select few decision-makers, notably beneficiaries and domain specialists. Furthermore, this knowledge is both incomplete and exhibits varying degrees of specificity. As a knowledge base should ideally comprise formal and explicit knowledge, substantial gaps exist between reality and the ideal state.

Furthermore, a CISD involves a collaborative effort among a set of actors to understand the problem, need, or business, identify weaknesses, convert them into opportunities, and elicit knowledge requirements from this intricate domain to propose suitable satisfaction alternatives. These characteristics make it particularly challenging to facilitate effective communication between actors in the CISD.

This communication surpasses the traditional understanding of interpersonal interaction,

involving spoken or written words, gestures, emotional expressions, or any other form modeling social behavior. It should be noted that a CISD, along with the high ambiguity and uncertainty that distinguish them, comprises different domains of knowledge. CISD knowledge, also known as metaknowledge, is independent of the domain and must be consciously managed by solution providers.

### 2.2 Systemic Thinking

Although its roots extend back decades, systems thinking remains foundational for comprehending and addressing complex situations through the analysis of interactions and relationships among the various elements of a system [19]. Before introducing the concept of systems thinking, it is noteworthy that systems thinking is an approach centered on comprehending the structure and behavior of systems, while systemic thinking is a perspective that acknowledges the interconnected and interdependent nature of elements within a system.

Both concepts underscore the significance of adopting a holistic viewpoint and grasping how different parts of a system mutually influence one another. The terms are occasionally used interchangeably, and the distinction might fluctuate according to the context and the preferences of different authors or practitioners. In this paper, we opt to use systemic thinking, represented by the acronym ST.

Thus, ST, fundamentally, conceives a system as an interconnected set of elements collaborating to achieve a common purpose. It surpasses the isolated analysis of individual components by prioritizing an overarching comprehension of the system, encompassing how its parts interrelate and mutually influence one another.

This perspective endeavors to elucidate the complexity of the world by considering it in terms of wholes and relationships rather than disassembling it into individual parts. Instead of concentrating solely on the components, attention is directed towards the interconnections and dynamics between them. Instead of solely focusing on the components, ST focus on the interconnection and dynamics between them.

According to Mardianto [20], the key principles of ST are the following:

a. Holism - the concept that a system must be viewed as a whole.
b. Inputs and outputs in a system - in a closed system, the input is determined once and is constant, while in an open system there are additional inputs that come from the environment.
c. Entropy - a unit for measuring abnormalities that exist in a system.
d. Hierarchy - something complex is made up of several smaller subsystems.
e. Goal seeking - a systemic interaction must have the same goal or end condition.
f. Regulation - feedback is much needed so that the system works as expected.
g. Equifinality - alternative ways to achieve the same goal.
h. Multifinality - achieve alternative goals from the same input.
i. Differentiation - specialized units have specialized functions as well.
j. Dualism - dual character system contradictory but very important for a system.
k. Modularity - separate or combine elements of the system according to the relationship.
l. Abstraction - the process of removing a characteristic of the system to define the basic characteristics.
m. Relation - the relationship between elements in a system.
n. Encapsulation - hide elements of the system.

ST has a rich tradition, and its effectiveness has been demonstrated since its introduction into the scientific field in diverse situations. Credited by Nakamori [21] as instrumental in the moon landing, ST has showcased transformative power across various fields such as management, engineering, ecology, health, and social sciences. Its contributions have fostered a more holistic understanding of dynamic systemics, solidifying its presence in the Cognitive Era.

Due to its holistic and interdisciplinary nature, numerous proposals have emerged, prompting a classification for improved comprehension [22, 23]. The most widely discussed classification, marked by opposing ideas and the necessity for varied

approaches, categorizes ST into hard and soft [22, 23, 24].

In its early application, ST primarily found its place in science and technology, particularly in structured domains. Here, problems were perceived as systemics with specific objectives, leading to the application of engineering methods to optimize and achieve established goals.

However, Checkland [22] argued that this approach lacked effectiveness in complex situations in the real world. In response, a paradigm shift in conceptualizing ST was proposed, giving rise to the soft proposal.

This perspective emphasizes the dimensions of human relations in the social sciences, with these ideas materializing in the Soft System Methodology (SSM).

### 2.2.1 Soft System Methodology

SSM is an approach designed to tackle complex problems in environments where a singular, clear-cut solution is elusive. It prioritizes understanding the perceptions and needs of involved individuals, employing flexible conceptual models to explore diverse perspectives and formulate solutions that are socially or politically acceptable and feasible.

In essence, SSM comprises several steps. Initially, it identifies the complex problem and analyzes the perspectives of those involved. Subsequently, a flexible conceptual model is crafted, providing a holistic representation of the situation. Various perspectives are then explored, leading to the generation of potential solutions. Ultimately, an agreement is reached on a solution deemed acceptable and feasible within the given context.

Three fundamental ideas underpin the SSM proposal [25]. Firstly, it discards the notion of addressing a real-world system in need of repair or improvement. Instead, it embraces the complexity and dynamism of the real world, proposing a systemic inquiry process that facilitates continuous learning cycles and suggests models of systemic activities within the problematic situation.

Secondly, acknowledging diverse viewpoints within the domain leads to the recognition that multiple models of activity systemics can be constructed, each offering its own perspective or Weltanshauung[1]. This necessitates exploring the systemic approach, proven effective in solving complex problems and supporting strategic decision-making [2].

Thirdly, the shift from addressing an "obvious problem" to understanding that a situation is considered problematic by individuals prompts the construction of various models for improvement. These models are recognized as conceptual ideas, not descriptions of real-world components, fostering a learning system that generates new knowledge about problematic situations. Ultimately, the selection of a feasible and desirable proposal requires consideration not only of the actor's viewpoint but also their unique history, culture, and aspirations.

In essence, when applied in complex domains, the soft system approach distinguishes itself by its capacity to confront challenges, discern patterns, and propose enduring satisfaction alternatives through a profound comprehension of underlying dynamics. This attribute positions it as a valuable tool for surmounting persistent challenges in complex domains.

The soft system approach embodies a perspective that endeavors to integrate the complexity of a domain, portraying it as a real-world entity interconnected in terms of wholeness and relationships, as opposed to oversimplifying, and fragmenting it into smaller parts. Leveraging this approach as a method facilitates the development of effective actions leading to satisfactory alternatives in complex domains.

Moreover, both approaches furnish an array of tools and methods for addressing intricate problems across various contexts, spanning from modeling dynamic behavior to engaging in critical reflection and strategic decision-making. Each approach exhibits distinctive characteristics and applications, and collectively, they constitute a

[1] "Weltanschauung" is a German term that translates to "worldview" in English. It refers to a comprehensive and fundamental perspective or outlook on the world, encompassing one's beliefs, values, attitudes, and understanding of reality.It also includes an individual's or a collective's philosophical, cultural, religious, and moral framework that shapes their interpretation of the world and guides their behavior. It provides a lens through which individuals or cultures perceive and make sense of the complexities of existence, morality, knowledge, and the overall human experience.

comprehensive toolkit for navigating complexity in decision-making and change management [23].

### 2.3 Knowledge Management

Knowledge Management (KM) is a set of processes, techniques and methods that involves the capture, storage, distribution, and effective application of knowledge within an organization or in any domain. It entails identifying critical knowledge for organizational success, facilitating its efficient sharing among team members, and leveraging it to continuously enhance processes and decision-making.

Establishing an organizational culture conducive to learning and collaboration is integral to KM [26]. This culture encourages viewing mistakes as learning opportunities, valuing knowledge sharing, and providing recognition for such contributions.

One of the most challenging aspects in KM pertains to the handling of tacit knowledge. Coined by Michael Polanyi in 1958 [27], tacit knowledge refers to "aspects of our knowledge that we cannot tell or much less describe precisely by writing down symbols." In essence, tacit knowledge encompasses intuitive or experiential knowledge that may not be easily articulated in words.

This form of knowledge is foundational across various domains, such as science, technology, and general learning, influencing actions and decisions in ways not always explicit. In the realm of KM, tacit knowledge assumes a pivotal role, representing the personal, subjective, and challenging-to-formalize knowledge held by individuals. Unlike explicit knowledge, which can be readily documented, tacit knowledge is rooted in people's experiences, values, and perceptions.

Although Polanyi laid the groundwork for tacit knowledge, it was only years later that Nonaka built upon his work, proposing a model of organizational knowledge creation [28]. The model's objective is to cultivate knowledge creation through processes of socialization, externalization, combination, and internalization. This systematic conversion of individual knowledge into organizational knowledge aims to generate sustainable competitive advantages.

The repercussions of tacit knowledge are not foreign to systemic thinking. According to Nakamori [21], the central tenet in ST, revolves around emergence, regardless of whether one adopts a soft or hard approach.

Emergent properties represent qualitative distinctions from the properties of any individual part within the system. Grasping emergent properties necessitates insight and intuition – an ability to synthesize systemic knowledge.

The integration of tacit knowledge underscores the capacity to amalgamate diverse fragments of knowledge, inductively infer a coherent whole, and extract new meanings.

Thus, harnessing tacit knowledge becomes imperative for enhancing decision-making, fostering innovation, and promoting organizational learning, especially in intricate and complex domains. Effectively managing tacit knowledge entails establishing mechanisms for sharing and transferring it among organizational members, fostering a culture that values and encourages the creation and dissemination of this form of knowledge.

#### 2.3.1 Knowledge Management on a Strategy for Requirements Engineering

In 2015, Olmos and Rodas [29] proposed leveraging the foundational concepts of Polanyi and Nonaka within a Knowledge Management on a Strategy for Requirements Engineering known as KMoS-RE. This approach aims to conceptualize and specify solutions in Informal Structured Domains (ISD) are domains where most of the knowledge is tacit and derived from expertise. Consequently, the definition of concepts and relationships between them is semi-informal and typically established through consensus.

The KMoS-RE process incorporates the five types of knowledge processes for complex problem-solving: Knowledge Elicitation, Knowledge Integration, Knowledge Application, Knowledge Validation, and Knowledge Exchange. These processes are amalgamated within a continuous and iterative cycle of model generation, verification, and validation until a solution that aligns with the decision-makers' preferences is specified, that is, make as many of the solution requirements explicit as possible.

KMoS-RE has demonstrated successful applications in various real-world scenarios [3, 29, 30, 31].

### 2.4 Fusing KMoS-RE and SSM

After exploring and experimenting with tools rooted in the philosophy of systemic thinking, the incorporation of SSM features into KMoS-RE was deemed advantageous. This involved establishing a foundational structure encompassing tools, methods, and processes to effectively tackle the intricacies of complex domains, as elaborated in section 3.

The idea of integrating ST into KM is not novel; Nakamori [21] advocates for collaborative efforts between the two disciplines. He contends that, while KM has predominantly focused on systemic approaches to knowledge, achieving innovation solely through KM is challenging.

On the contrary, ST, specialized in addressing complex problems, encounters formidable challenges when dealing with complex domains and endeavoring to manage knowledge pieces akin to KM practices.

Why is it advocated to fuse SSM and KMoS-RE? Like every tangible and intangible technological artifact devised by humans, both proposals have their advantages and disadvantages and are subject to improvement. KMoS-RE was specifically designed for eliciting knowledge requirements, so the artifacts it proposes and the cycle on which it is based are primarily conceptualized to convert as much tacit knowledge into explicit knowledge.

However, it does not have sufficient mechanisms to deal with complex domains, such as the management of uncertainty, the identification of alternatives, and the management of interdisciplinary work groups. Regarding the soft systems methodology, it does not have the mechanisms to deal with knowledge management, particularly with the challenges presented by tacit knowledge.

As a result, a strategic framework named Knowledge Management on a Strategy options through Soft Systemic Analysis (KMoS-SSA) has been devised [4]. This framework not only acquires pieces of knowledge but operates within a continuous cycle of knowledge enrichment. It involves all the actors involved in the project, from domain specialists, decision makers and solution providers. By representing the complex domain through models that incorporate diverse perspectives, KMoS-SSA supports the continuous reflection of all actors involved. It is about obtaining, validating, discussing, and sharing knowledge to achieve solutions that are not only efficient but also desirable and satisfactory.

## 3 Methodological Framework KMoS-SSA

KMoS-SSA is a methodological Framework that provides a systematic and knowledge-based approach to solving problems in Complex Informal Structured Domains (CISD). KMoS-SSA uses qualitative, rational, interpretive, and cognitive techniques and methods to generate a cognitive architecture, which can be defined as a set of artifacts composed of symbolic representations of knowledge that provide the semantic basis of the domain and allow interpreting, defining, and exploring the various alternatives of a domain to its problems and needs under scrutiny [2].

KmoS-SSA promotes the elicitation and enrichment of knowledge through debate, learning and understanding to evolve complex domains. It consists of the Knowledge enrichment cycle and the KMoS-SSA process model.

### 3.1 Knowledge Enrichment Cycle

Knowledge enrichment refers to the process of improving or expanding the depth, breadth, or quality of existing knowledge. This may involve adding, modifying, or even discarding information, context, or relationships to existing knowledge to make it more valuable, insightful, or useful.

The knowledge enrichment cycle is depicted as a cyclic process of symmetric loops converging at the center, forming an infinite horizontal structure, as illustrated in Fig. 2. This cycle involves two sets of fuzzy boundary stages: those aligned with System Thinking and those associated with real-world situations.

While the cognitive architect typically coordinates the actions in this cycle, active participation is encouraged from domain specialists, solution providers, and decision-makers. These actions revolve around acquiring, expanding, and consistently refining domain

knowledge to underpin the conceptualization and construction of problem-solving alternatives.

The continuous engagement in the cycle contributes to the expansion and deepening of domain understanding. Although the set of stages that make up the cycle can vary based on the characteristics of the CISD under analysis, a fundamental set of stages would typically include:

- Knowledge Elicitation (KE): Knowledge elicitation is a stage where a systemic process, a series of interconnected and interdependent activities or steps that work together to achieve a particular goal or outcome within a system, is carried out involving the identification, acquisition, organization, and application of pertinent information to comprehend a specific domain. The relevant information, constituting pieces of knowledge, emanates from diverse sources, including domain specialists, knowledge users, customers, beneficiaries, and other actors within the domain. This process is imperative for a comprehensive understanding of beneficiaries' needs and expectations, ensuring that any proposed solution aligns with their knowledge and functional requirements. Typically, the process initiates with a clear identification of the domain to establish the limits and scope of the knowledge area to be acquired.

  Key actors, encompassing relevant individuals, groups, or entities in the domain, are identified, along with their roles and perspectives. The process involves the meticulous identification, analysis, and evaluation of internal and external information sources, considering factors such as reliability, relevance, and timeliness. Strategies for information gathering and knowledge piece development are formulated, followed by data collection and analysis.

  Tacit Knowledge is elicited through interviews or interactions with Domain Specialists. In subsequent stages of the cycle, knowledge modeling is undertaken, employing conceptual models or visual representations of the obtained knowledge. Validation and feedback mechanisms are employed to verify the integrity and coherence of the acquired knowledge, with Domain Specialists and key actors providing feedback to rectify potential errors. Documentation ensues, organizing the identified knowledge in an accessible and structured manner.

  The application and transfer of knowledge involves implementing acquired knowledge in practical domains and facilitating knowledge transfer through training and effective communication. Continuous evaluation is integral, involving the monitoring of knowledge-based applications' performance and the adaptive evolution of knowledge in response to changes in the domain.

  This process is iterative and dynamic, recognizing that knowledge evolves over time and through experience. Active collaboration with Domain Specialists and continuous feedback are indispensable components of an effective knowledge elicitation process. Finally, it is noteworthy that the knowledge enrichment cycle commences with this stage.

- Knowledge Enrichment (KEnr): This stage involves a thorough reflection and validation of the collected pieces of knowledge, evaluating their connection to the problem situation from each domain specialist's perspective. A critical assessment is conducted, addressing both the quality and relevance of the gathered information. Moreover, an organizational framework is established, offering a foundational structure for understanding relationships and patterns, with the goal of constructing alternatives based on validated knowledge fragments. This stage necessitates continuous exploration and the ongoing acquisition of knowledge, ensuring that all actors remain focused on the problem situation and its related issues, thereby maintaining clear and unambiguous definitions. It also involves a continuous search for and acquisition of pertinent information from various sources. These actions are carried out, both to enrich the knowledge of the application domain of the DS, as well as that of the solution domain, with the solution providers and decision makers.

- Model Generation (MGen): This stage encompasses the execution of artefacts or models to authenticate and strengthen knowledge. Employing suitable modeling tools is essential for producing artifacts that effectively represent a domain and potential solutions. Among these models are linguistic models, conceptual models, strategic objective models, knowledge matrices, pertinent systems models, etc. It also includes systems thinking techniques such as a discussion, and reflection guide (PQR), a tool facilitating the comprehension of various dimensions and considerations in the analysis (CATWOE), and root definitions.
- Model Discussion (MDisc): The model discussion stage is in the central part of the cycle, considering it is the union of the real world with ST. It is one of the most important stages in which the models are explained to the Domain Specialists, this knowledge made explicit, not only serves to validate that the content of the models represents in the most correct way possible the reality of the domain, but also it is essential to provoke reflection in the Domain Specialist about their vision of the world, which would allow them to modify their mental scheme. Likewise, the Domain Specialist could observe the knowledge of other Domain Specialist and together make decisions regarding the structure of the domain and possible alternative solutions.
- Model Validation (MVal): In this stage, Domain Specialists meticulously evaluates the model's alignment with reality, providing critical reflections and offering feedback. The validation process centers on ensuring that the artifact fulfills the requirements and expectations of the beneficiary concerning their problem situation, comprehending how it effectively addresses that situation. At this juncture, the cognitive architect coordinates a discussion involving all domain actors, including essential Domain Specialists and decision-makers. During this phase, results and experiences are analyzed, feedback from specialists is gathered, and the necessity to incorporate additional Domain Specialists or adjust the participation of some is assessed. Identified shortcomings pinpoint areas requiring deeper understanding. Artefacts are refined and enhanced, modifying concepts or approaches based on the reflection and feedback received. Finally, a comprehensive check ensures that the generated artifacts meet the requirements identified in the Knowledge Enrichment stage.

It is important to highlight that each stage involves actions geared towards facilitating knowledge gathering, communication, and sharing. These actions encompass activities aimed at comprehending the CISD, such as conducting interviews, transcribing interviews, and identifying/selecting internal and external information sources.

These activities contribute to a thorough description of the domain, encompassing elements such as actors, roles, values, norms, symbols, requirements, concepts, and relationships inherent to the domain. Ensuring clear and precise communication of concepts and findings is crucial, sharing all validated pieces of knowledge with every participant in the domain. This practice fortifies collective knowledge through presentations, publications, or interactions among all participants. Knowledge acquisition is achieved by pinpointing the problem situation and associated issues, ensuring a clear and unambiguous definition. It also involves actively seeking and obtaining pertinent information from diverse sources.

It is essential to bear in mind that the cycles typically maintain this structure because they consistently revert to the beginning of the cycle but start from the updated collective knowledge. Hence, it is an iterative knowledge enrichment cycle, wherein continuous learning implies ongoing enhancements and adaptations. The ability to identify unknowns, apply new knowledge, and share discoveries is fundamental for an effective knowledge enrichment cycle.

Various techniques and tools can be employed for the diverse actions at each stage, and the selection depends on the specific characteristics of

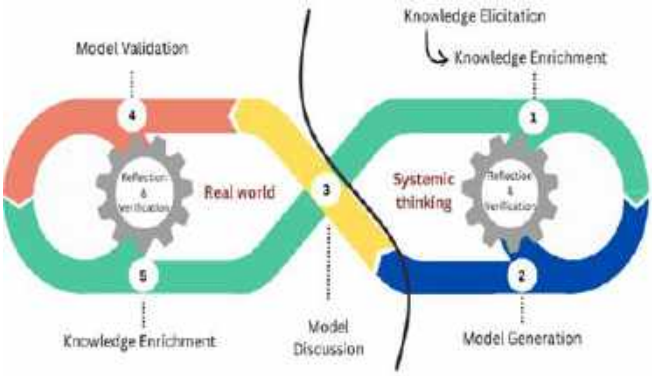



**Fig. 2** Knowledge Enrichment Cycle: The illustration depicts a cyclical process of symmetrical loops converging at the center, forming an infinite horizontal structure

the problem and its domain. For instance, a detailed analysis of the relationships between system components is useful in identifying interconnections.

Visual tools like concept maps or influence diagrams can aid in visualizing these interconnections. Modeling feedback loops through visual representations helps explore how changes in one area affect other areas. Diagrams illustrating feedback loops are valuable for understanding how actions impact the system over time, differentiating between positive (change) and negative (no change) feedback to grasp amplification or stabilization effects.

This cycle comprises two sets of stages with fuzzy boundaries: those aligned with Systemic Thinking and those connected to real-world. The active stage within the cycle is identified by numbered circles and associated colors. It is crucial to emphasize that the knowledge enrichment stage is incorporated into both cycles, and the model discussion stage intricately interconnects them. Furthermore, reflection and validation activities are systematically conducted in both cycles. Source: Own creation.

Recognition of emergencies involves collecting and categorizing emergent behaviors, continually monitoring unexpected outcomes, and adapting to changes not directly attributed to individual elements. For domain considerations, analyzing contextual factors influencing the system, exploring complex and non-linear causal relationships, and using contextual maps to visualize the domain's breadth are desirable.

To support the holistic approach and distinguish patterns, data analysis and observation can aid in identifying recurring patterns. Implementing feedback systems for result evaluation and establishing continuous organizational learning processes is useful for feedback and continuous learning. Pursuing systemic change involves conducting root cause analyses to address problems at their source, proposing significant changes in domain structures and processes, and designing and implementing strategic interventions to achieve sustainable improvements. These actions are typically executed in an iterative and collaborative manner, engaging various actors, and employing specific tools based on the requirements of the systemic analysis.

### 3.2 KMoS-SSA Process Model

This subsection provides a schematic overview of the steps and actions required to integrate Systemic Thinking and KM. This involves developing a structured process model that takes as a basis the Knowledge Enrichment Cycle and that incorporates the principles and tools of both methodologies: SSM and KMoS-RE.

It is noteworthy that the realization of these models involves the application of KMoS-RE techniques, specifically employing KM to represent information through models and imparting structure to the acquired knowledge. KMoS-RE is augmented with a systemic approach to comprehend the domain from a broader

perspective, encompassing the interconnections within the CISD. Strategic objectives are modeled and synchronized with desirable goals, thereby ensuring the feasibility and desirability of solutions. Consistent alignment is maintained across various KMoS-RE modeling stages, including knowledge modeling and the modeling of functional aspects of the proposed solution, with systemic principles.

The components of the KMoS-SSA process model are outlined below, following the sequence depicted in Fig. 3, guiding from the "start" point to the "end" point:

1. Initial Evaluation, along with subsequent actions, discerns the complex domain or problem situation necessitating intervention through knowledge elicitation tasks. Examine the characteristics of the domain to ascertain the existence of tacit knowledge, ambiguity, and uncertainty.

   Identify actors, encompassing Domain Specialists, beneficiaries, stakeholders, users, clients, and other pivotal participants. Grasp their viewpoints, roles, and contributions within the context. Additionally, pinpoint pertinent information sources and initiate initial exercises for constructing a domain structure. The initial steps of the scheme are depicted in the central section of Fig. 3, featuring the "start" indicator followed by the Initial Evaluation and Information Source Identification tasks. These tasks are taken from KMoS-RE.
2. Domain Framing and Conceptualization: The problematic situation is framed and conceptualized, considering social, cultural, and political aspects, along with a profile of domain specialists. Once the models or artifacts are generated, the cognitive architect must decide if it is appropriate to initiate the validation process with domain specialists.

   This decision should be based on the architect's judgment of having gathered enough material for validation. These tasks are adapted from analyzes 1, 2 and 3 of the SSM.
3. The cognitive architect will consider validating the artifacts or models to determine their approach to the real world, a task that must be carried out with domain specialists.
4. Models are deliberated upon during the Model Discussion (MDisc) stage, illustrating the problem, knowledge pertaining to it, and relationships within the CISD, culminating in their validation at the Model Validation (MVal) stage.
5. In the context of operating within CISD, the recurring tasks of reflection and verification (Sections of the fig. 3 are depicted at both the far-right center and far-left center), integral components of this framework, assume paramount significance. These tasks play a pivotal role in ensuring the precision and accuracy of both the information and models devised. Furthermore, they scrutinize the comprehension of the intricate domain and the alignment of the proposed alternatives with the authentic situational context. Ambiguity is systematically mitigated through these processes, offering actors the opportunity to reassess assumptions, interpretations, and models.

   Verification establishes the reliability and validity of information utilized in decision-making processes. This approach fosters a culture of continual learning, encouraging actors to review experiences, discern valuable insights, and adapt their methodologies accordingly. By learning from errors or misunderstandings, verification contributes to the progressive refinement of the overall problem-solving methodology. Confidence among actors, including specialists and beneficiaries, is reinforced as they seek assurance in the sound foundation of the generated alternatives.

   Verification functions as a testament to the credibility of the acquired information and models, thereby instilling confidence in the decision-making process. Moreover, these processes substantiate decision-making by allowing actors to critically assess proposed alternatives, consider alternative perspectives, and evaluate potential impacts. Verification ensures that decisions are anchored in

validated and reliable information, thus facilitating more enlightened and effective decision-making outcomes. The iterative refinement of models, strategies, and actions is facilitated through the assimilation of lessons derived from reflection and the insights gleaned through verification.

Compliance with quality standards is streamlined, particularly in domains where reflection and verification assume a pivotal role in meeting prescribed standards. Furthermore, these processes contribute to consensus-building by transparently reviewing information and models.

This transparency enables actors to align their perspectives, thereby mitigating conflicts and fostering improved collaboration. At the edges of Fig. 3, it is illustrated the actors involved in this process. On the left side, representing the real world, domain specialists are responsible for carrying it out.

On the right side, representing systemic thinking, solution providers take on this role. In KMoS-RE this task is only considered for solution providers. The systemic approach is committed to an analysis of alternatives, which implies more in-depth involvement of domain specialists.

6. The knowledge enrichment stage holds paramount significance when operating within CISD, where decision-making concerning alternative solutions is of utmost importance. Through this stage, the quantity and quality of knowledge pieces related to potential solution alternatives are increased, erroneous knowledge is eliminated, contributing to enhanced comprehension, reduced uncertainties, alternative identification, and support for continuous improvement.

   Consequently, more robust, and effective decisions are facilitated. Thus, the undertaking of knowledge enrichment before and after deciding upon a solution alternative is deemed essential to guarantee that decisions are well-informed, effective, and adaptive to the intricacies presented in a CISD.

   This task can be predominantly performed by domain specialists (6a) or by solution providers (6b). This state is taken from KMoS-RE.
7. In CISD, a wealth of data and information, constituting valuable knowledge, is present. This data and information may originate from explicit sources within or external to the domain. Therefore, the analysis of data, information, knowledge, experience, and scientific literature (top center of Fig. 3) is deemed indispensable.

   This analytical process establishes a robust foundation for decision-making, diminishes uncertainties, optimizes resource utilization, cultivates innovation, and facilitates continuous improvement within complex contexts.

   The analysis, particularly conducive to informed decision-making, pattern identification, uncertainty reduction, resource optimization, enhanced learning, support for information and knowledge validation, and potential innovation stimulation, plays a pivotal role in these domains. This state is not explicitly considered, neither in KMoS-RE, nor in SSM. The model proposed by Nakamori [20] was considered for its inclusion.
8. The model generation stage creates a representation, albeit partial, of a potential structure within a CISD. This involves capturing the key components, relationships, and dynamics of the domain (MGen stage located in the center of Fig. 3). This process facilitates a more comprehensive understanding of the intricate nature of the domain, enabling actors to gain improved insight into the problem or situation at hand.

   A model serves as a visual or conceptual framework, aiding in the communication of complex ideas and the identification of interconnections among various elements within the domain. This fosters collaboration and shared understanding among actors, including specialists, beneficiaries, and decision-makers.

   Furthermore, a well-constructed model contributes to the information analysis task by helping to identify patterns, trends, and

potential interdependencies within the CISD. Participation in the generation of a model not only contributes to the analysis of information but also encourages actors to articulate their tacit knowledge, assumptions, and mental models.

This, in turn, fosters a shared learning experience. The model generation process provides a platform for actors to collectively reflect on their perspectives and views, promoting a deeper and more nuanced understanding of the domain. It is proposed that models used in both KMoS-RE and SSM be considered.

From KMoS-RE the linguistic model, the conceptual model, the process model, among others, would be taken. From SSM the Big Picture, the strategic goals model, among others, would be taken.

9. In this context, the determination of whether a solution is satisfactory (upper central diamond in Fig. 3) involves a comprehensive assessment that encompasses various factors within the CISD.

   Several factors are related to understanding actors' perspectives, ensuring continuous knowledge enrichment, validating, and thoroughly testing models and alternatives through debates and reflection, aligning with beneficiaries' strategic objectives, making well-informed decisions, facilitating consensus building, and more.

   By considering these aspects, it can be collectively determined by actors whether a proposed solution adequately addresses the complexities and challenges presented by the CISD.

The integration of SSM into KMoS-RE within the methodological framework KMoS-SSA underscores a holistic approach to knowledge management in CISD.

This approach comprehensively considers both structured and unstructured elements, considering the social and cultural dimensions of the problem situation. This process is demanding, requiring the mapping, understanding, and structuring of different perspectives related to the phenomenon, particularly those related to the decision-making processes influencing the consideration of alternatives to address the situation and achieve a certain level of satisfaction.

The process involves all actors, stakeholders, or beneficiaries in the CISD, sources of information and relevant entities or objects providing assessments for decision-making. It is essential to recognize that the application of this process may require a significant investment of time and resources, with the primary objective being the production of sound and rational processes that can be effectively implemented.

## 4 Analysis of the Integration of Knowledge Management and Systems Thinking

The KM conducted by the KMoS-RE process to elicit pieces of knowledge, ISD models, or knowledge requirements involves systemic actions that consider the interconnected and dynamic nature inherent to CISD. Essentially, the KM within CISD through the methodological strategic framework KMoS-SSA, which comprehensively integrates the entire KMoS-RE, can be defined as follows:

1. Definition of the CISD and scope: Delineate and identify the CISD under analysis. Establish the system's boundaries and context, considering the interconnections and relationships between domains.
2. Identification of CISD actors: Identify and categorize the key actors involved in the CISD, considering their perspectives and roles within the system.
3. Problems and needs analysis: Conduct a thorough analysis of problems and needs within the domain. Utilize techniques such as interviews, surveys, and document analysis to gather information and knowledge on existing challenges.
4. Recognition of interconnections: Analyze the interconnections between different elements of the system. Identify how changes in one part of the system may impact other areas.

ISSN 2007-9737



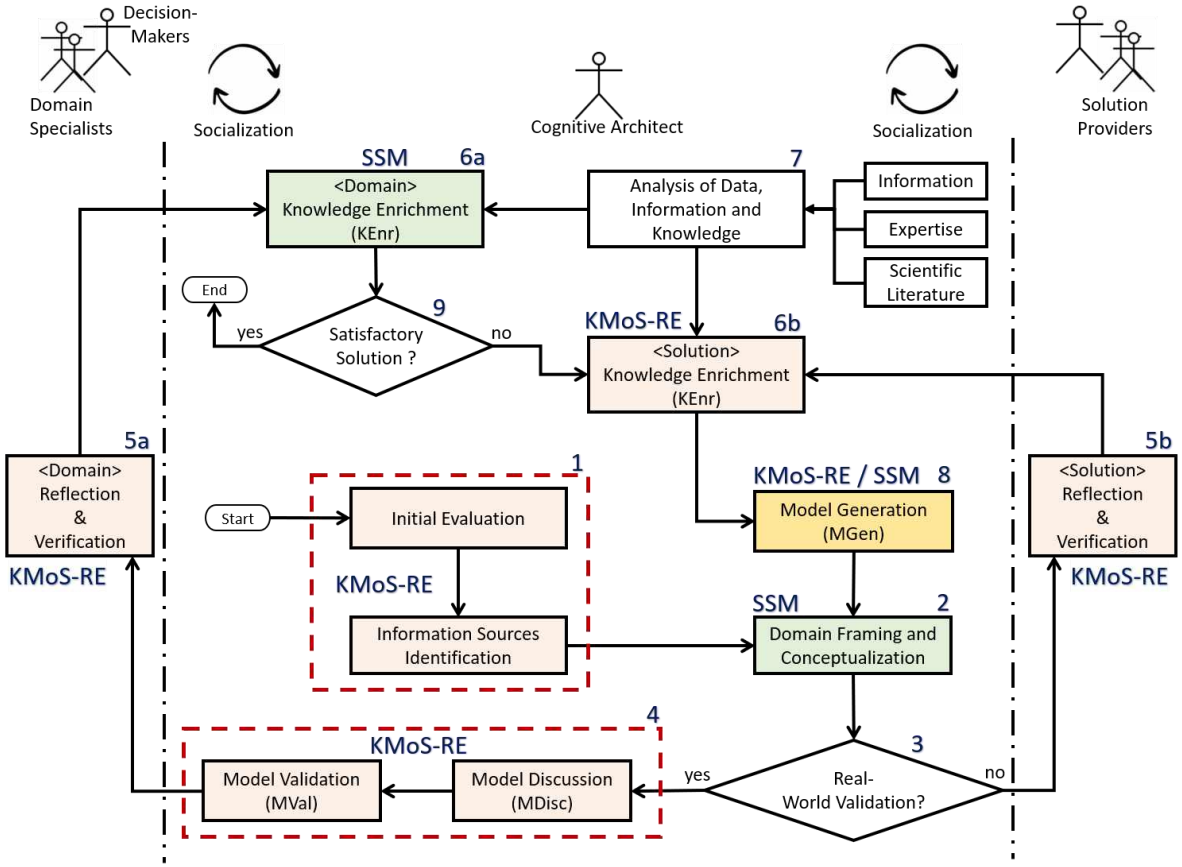



**Fig. 3** Depicts the KMoS-SSA framework, illustrating the stages and tasks essential for achieving KM. Rectangular boxes symbolize the stages and their associated tasks, while diamonds represent decisions. “Start” and “End” points are denoted by ellipses. At the top, human figures symbolize the principal actors within a CISD and their interrelationships. Source: Adapted from [4]

5. Requirements elicitation: Employ methods such as interviews, workshops, and questionnaires to elicit requirements from decision-makers and relevant actors in the CISD. Encourage the participation of decision-makers to understand their expectations and needs.
6. Contingency and scenario analysis: Explore various scenarios and situations to understand contingencies and possible variations in domain behavior. Identify requirements that address all potential scenarios within the CISD.
7. Modeling the CISD through Domain Knowledge: Employ modeling tools to visually represent pieces of knowledge, requirements, etc. through images, documented descriptions, use case diagrams, flowcharts, and conceptual models. The objective is to enhance communication and ensure comprehension among all actors within the domain.
8. Documentation of the elicitation of pieces of knowledge and their impact on the CISD: Produce comprehensive documentation that includes all identified pieces of knowledge, their priorities, and associated constraints for actors within the CISD.
9. Change management: Establish a process for managing changes to requirements over time. Consider the ongoing evolution of the CISD and its impact on requirements.

The elucidates that the process of eliciting pieces of knowledge, requirements, and domain modeling navigates the intrinsic complexity of CISD by acknowledging the multiple dimensions, interconnections, and dynamics inherent in these environments. In conclusion, active collaboration with decision-makers and a commitment to adaptability amid continuous change are fundamental aspects of this approach.

## 5 Conclusions and Future Work

The incorporation of systems approaches, specifically SSM interventions, has garnered significant attention in addressing real-world situations characterized by intricate interrelationships, diverse perspectives, and the management of power relations in the context of the Cognitive Era. This integration aims to facilitate and embrace systemic KM. Conventional solutions may prove inadequate in addressing such complexity, prompting the need for innovative and adaptive approaches, including the integration of cognitive technologies such as AI.

One of the foremost challenges lies in effectively managing the extensive TK derived from experience to address the multifaceted aspects of work-related stress or any other CISD. Consequently, KM emerges as a pivotal strategy for proposing alternative solutions, particularly within the context of a CISD.

Historically, both systemic approach tactics and KM tactics have encountered notable challenges when individually applied to domains such as environmental and business management [21], social and mental health issues [29,30,31,32], sustainability, and others [33]. Consequently, these challenges endure, owing to the inherent complexity of such issues. The convergence and harmonization of perspectives, worldviews, and knowledge management, among other factors, present complexities that demand the formulation of integrated philosophies or approaches to effectively address CISD.

It is important to acknowledge that CISDs constitute an inherent aspect of everyday life. These domains extend beyond specific instances such as work-related stress, ergonomics management in supply chains, gender studies in industry, or education. Instead, CISDs permeate all contexts in which individuals utilize pieces of knowledge for decision-making and problem-solving. Consequently, frameworks like the proposed KMoS-SSA hold promise in effectively addressing these domains, whether they manifest on an industrial or institutional scale, at a local or global level, or involve processes of cognitive transformation.

In conclusion, the current study employs the KMoS-SSA framework to offer practical solutions for institutions, organizations, or companies aiming to alleviate the impacts of issues within a CISD. Specifically, in the domain of work-related stress, where initial positive outcomes of KMoS-SSA are emerging, the framework significantly contributes to the ongoing dialogue regarding effective stress management strategies. Currently, we are working on a detailed analysis of the artifacts or models that will be carried out in each stage of the process model.

As future work, we intend to support the maquiladora industry in intelligent solutions to support them in making decisions to mitigate the effects of work stress and enable them to comply with the corresponding regulations.

## Acknowledgments

On behalf of all co-authors and collaborators, we express our most sincere gratitude and eternal admiration for Dr. Jazmín Georgina Licona Olmos. Her contribution was invaluable, and her legacy remains on these pages. She was a brilliant researcher, an exceptional mentor, and an inspiring person.

## References

1. **Jiménez-Galina, A. M., Maldonado-Macías, A. A., Olmos-Sanchez, K. M., Hernández, I., Estrada-Saldaña, F., Vázquez-Gálvez, F. A. (2024).** Framework for heterogeneous data management: An application case in a NoSQL environment from a climatological center. Computación y Sistemas, Vol. 28, No. 1, pp. 167–178. DOI: 10.13053/CYS-28-1-4474.

2. **Rodas-Osollo, J. (2023).** An interesting adventure accompanied by CMCg.I model. Zenodo & Latin American Institute of Critical Pedagogy. DOI: 10.5281/zenodo.10111223.
3. **Rodas-Osollo, J., Olmos-Sanchez, K., Portillo-Pizaña, E., Martínez-Pérez, A., Alemán-Meza, B. (2021).** An archetype of cognitive innovation as support for the development of cognitive solutions in smart cities. Innovative Applications in Smart Cities. CRC Press, 1st Edition, pp.89–105. DOI: 10.1201/9781003191148.
4. **Rodas-Osollo, J., Olmos-Sanchez, K., Kotlyarova, I. (2024).** A conceptual framework–KMoS-SSA: synergizing quality management and technology. 25th international symposium Quality-Yesterday, Today, Tomorrow, Croatian Quality Managers Society, ˇSibenik, Croatia, Vol. 25, No. 1.
5. **Nikhlis, N., Iriani, A., Hartomo, K. D. (2020).** Soft system methodology (SSM) analysis to increase the number of prospective students. INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi, Vol. 4, No. 1, pp. 63–74. DOI: 10.29407/intensif.v4i1.13552.
6. **Rohajawati, S., Fairus, S., Saragih, H., Akbar, H., Rahayu, P. (2021).** A combining method for systems requirement of knowledge-based medical hazardous waste. TEM Journal, Vol. 10, No. 4, p. 1761. DOI: 10.18421/TEM104-37.
7. **Zahid, A., Sharma, R., Wingreen, S., Inthiran, A. (2022).** Soft systems modelling of design artefacts for blockchain-enabled precision healthcare as a service. ICEB 2022 Proceedings (Bangkok, Thailand), Vol. 22, pp. 451–467.
8. **Hadi, A. H., Pramuhadi, G., Susantyo, B., Wahyono, E. (2023).** Sustainability concept design of Robusta coffee agroindustry Kalibaru with soft system and decisions support system methods. International Journal of Sustainable Development & Planning, Vol. 18, No. 5. DOI: 10.18280/ijsdp.180504.
9. **Larson, E. J. (2021).** The myth of artificial intelligence: Why computers can't think the way we do. Harvard University Press. DOI: 10.4159/9780674259935.
10. **Forastieri V. (2016).** Workplace stress 'a collective challenge' as work-life boundaries become blurred. UN NEWS. https://news.un.org/en/story/2016/04/527952.
11. **Karasek, R. A. (1979).** Job demands, job decision latitude, and mental strain: implications for job redesign. Administrative Science Quarterly, Vol. 24, No. 2, pp. 285–308. DOI: 10.2307/2392498.
12. **Siegrist, J. (1996).** Adverse health effects of high-effort/low-reward conditions. Journal of Occupational Health Psychology, Vol. 1, No. 1, pp. 27–41. DOI: 10.1037/1076-8998.1.1.27.
13. **House, J. S. (1981).** Work stress and social support. Addison-Wesley.
14. **Hobfoll, S. E. (1989).** Conservation of resources: A new attempt at conceptualizing stress. American Psychological Association, Vol. 44, No. 3, pp. 513–524. DOI: 10.1037/0003-066X.44.3.513.
15. **Lazarus, R. S., Folkman, S. (1984).** Stress, appraisal, and coping. Springer.
16. **Ochoa-Zezzatti, A., Mejia, J., Diaz, J., Sánchez-Solís, P., García, V., Rivera, G., Florencia-Juárez, R. (2021).** Analysis of mental fatigue under delivery pressure and considering creativity and precision to organize and distribute a diorama to represent social issues based on cultural algorithms. Technological and Industrial Applications Associated with Intelligent Logistics, pp. 405–416. DOI: 10.1007/978-3-030-68655-0_20.
17. **STPS (2016).** Bienestar emocional y desarrollo humano en el trabajo: Evolución y desafíos en México.
18. **STPS (2018),** NOM-035-STPS-2018. DOF. https://dof.gob.mx/nota_detalle.php?codigo=5541828&fecha=23%2F10%2F2018#gsc.tab=0
19. **Forrester, J. W. (1997).** Industrial dynamics. Journal of the Operational Research Society, Vol. 48, No. 10, pp. 1037–1041. DOI: 10.1057/palgrave.jors.2600946.
20. **Mardianto, M., Ahyar, S., Abidin, Z. (2022).** Basis and principles of systematic thinking in

education. Edumaspul, Vol. 6, No. 2, pp. 2058–2062. DOI: 10.33487/edumaspul.v6i2.3746.

21. **Nakamori, Y. (2020).** Fusing systems thinking with knowledge management. Journal of Systems Science and Systems Engineering, Vol. 29, No. 3, pp. 291–305. DOI: 10.1007/s11518-019-5450-8.
22. **Reynolds, M., Holwell, S. (2020).** Systems approaches to making change: A practical guide. Second Edition, Springer, London, pp. 201–253. DOI: 10.1007/978-1-4471-7472-1.
23. **Hanafizadeh, P., Mehrabioun, M. (2022).** The nature of hard and soft problems and their problem-solving perspectives. Journal of Systems Thinking in Practice, Vol. 1, No. 3, pp. 22–48.
24. **Checkland, P. B. (1989).** Soft systems methodology. Human systems management, Vol. 8, No. 4, pp. 273–289. DOI: 10.3233/HSM-1989-8405.
25. **Anzures-García, M., Sánchez-Gálvez, L. A., Hornos, M. J., Paderewski-Rodríguez, P. (2018).** A workflow ontology to support knowledge management in a group's organizational structure. Computación y Sistemas, Vol. 22, No. 1, pp. 163–178. DOI: 10.13053/cys-22-1-2781.
26. **Polanyi, M. (1958).** Personal knowledge, No. 2. University of Chicago. isi:A1959CCP2000015.
27. **Nonaka, I., Takeuchi, H. (1995).** The knowledge-creating company: how japanese companies create the dynamics of innovation. Hardvard Business Review, pp. 96–104.
28. **Olmos-Sanchez, K., Rodas-Osollo, J. (2016).** A strategy of requirements engineering for informally structured domains. International Journal of Combinatorial Optimization Problems and Informatics. Vol. 7, No. 2, pp. 49–56.
29. **Olmos-Sánchez, K., Rodas-Osollo, J. (2020).** Helping organizations manage the innovation process to join the Cognitive era. 2020 8th International Conference in Software Engineering Research and Innovation (CONISOFT), pp. 1–10. IEEE. DOI: 10.1109/CONISOFT50191.2020.00012.
30. **Rodas-Osollo, J., Olmos-Sanchez, K. (2022).** Toward optimization of medical therapies with a little help from knowledge management. Recent Advances in Knowledge Management, IntechOpen. DOI: 10.5772/intechopen.101987.
31. **Trochim, W. M., Cabrera, D. A., Milstein, B., Gallagher, R. S., Leischow, S. J. (2006).** Practical challenges of systems thinking and modeling in public health. American journal of public health, Vol. 96, No. 3, pp. 538–546. DOI: 10.2105/AJPH.2005.066001.
32. **Tona, O., Asatiani, A. (2023).** Designing digital solutions for sustainability: Navigating conflicting stakeholder requirements with dignity in mind. Journal of Information Technology Teaching DOI: 10.1177/20438869231216995.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Karla M. Olmos Sánchez.*

ISSN 2007-9737

# Graph-based Saliency Detection: Integrating Multilevel Features and Analyzing the Distraction Using DNN

P Kokila[1,*], R Kanagavalli[2]

[1] VTU, Belagavi, Karnataka, Department of CSE, PESU,
Bengaluru, Karnataka,
India

[2] The Oxford College of Engineering,
Department of Information Science and Engineering,
India

{kokila.p.anand, kanaga.ksr}@gmail.com

**Abstract.** Salient object detection is a key component in a number of computer vision applications, including scene understanding, object recognition, and image understanding. In salient object detection tasks, deep neural network models have become more popular recently, and have displayed astounding performance. But there is definitely potential for development, especially in terms of capturing multi-level characteristics and dealing with distractions in complicated environments. We suggest a fully convolutional neural network (FCN) with a dilation kernel to resolve the time-consuming shortcoming of conventional CNN, this article seeks to improve salient object detection where one attempt will be made to map the raw input to a dense spatial map and due to this, complex and low-contrast images are also able to give useful information. Distraction analysis finds the parts of an input image that interfere with the saliency detecting process. As a way to determine the saliency value of each node by supplying a coarse saliency map, the graph architecture allows to define the node with various edges. The boundary of the salient item is accurately highlighted using the active contour refinement method. On six well-known public data sets and nine cutting-edge saliency detection techniques, extensive trials are run. In relation to accuracy and robustness, our framework outperforms competitors in a variety of challenging situations, according to the results.

**Keywords.** SOD, dilated kernel, dense labeling, distraction mining, graph construction, active contour estimation.

## 1 Introduction

Saliency detection in computer vision concentrates on identifying the eloquent and visually dominating region in an image. The objective of salient object detection is to quickly recognize and distinguish the most prominent region in an image just like the human visual system [1]. The human visual system could be imitated technically by the model of visual attention, it is a computational method intended to accomplish the goal of the human visual system, i.e. it involves simulating human eye fixation on a certain area by ignoring other objects.

The salient attention model is used in a wide area of the field like machine intelligence, cognitive psychology, and robotics to increase the effectiveness of visual processing [2]. Its application has not stopped till that, Salient object detection, in contrast is more appropriate for a wide range of computer vision activities [3] including semantic segmentation, object location and detection, content-aware image modification, visual tracking, and human reidentification [4, 5].

It has a wide range of applications because it is more focused on maintaining the integrity of the probable object. Although [6] numerous top-notch models have been released, salient object detection and recognition are still difficult

because of a number of intricate elements that are present in everyday circumstances [7].

According to perceptual research [8, 9], the fundamental factor that affects saliency is visual contrast, it's been effectively proposed in a number of multimodal salient object recognition techniques had built through either global or local contrast modelling [10, 11, 12].

Local contrast models, for instance, can have trouble effectively separating out sizable homogeneous zones inside conspicuous objects, whereas global contrast information may struggle to deal with images with a complex background. Despite the fact that salient object detection is a task for machine learning-based techniques do exist [13, 14, 15], they are mostly focused on merging several saliency maps calculated by various techniques [16].

Deep convolutional neural networks (DCNN) have recently gained popularity in salient object detection due to their potent feature representation, and significantly improved performance over the old techniques [17]. Deep CNN-based techniques inference and training are often carried out region-wise methods, first, a section of an image is split into set of regions or patches.

However, this leads to significant storage and computational redundancies, which require extensive training and testing time, saving the deep features recovered from a single image requires hundreds of megabytes of storage [18]. This limitation motivated us to develop a CNN model using dense computing, and it provides a facility to transfer the input image of any size to a saliency map of the same size. [19] has mentioned several saliency map detection and edge detection using deep learning techniques.

In SOD preserving the outline of the object is so important, it can be attained by simple fully convolutional networks (FCNs), but the result of FCN is mostly a hazy outline because pixel-level correlation is normally not taken into account [20]. So, to learn and to map a complete image to a saliency prediction with pixel-level accuracy, we design a multiscale FCN in the first stream. By extracting visual contrast information from multiscale receptive fields, in addition to learning multiscale feature representations, our FCN can accurately evaluate the saliency of each pixel [21]. The following contributions are made by this paper.

1 To derive a dense saliency prediction straight from the raw input image in a single forward pass, a multiscale VGG-16 [22] network pre-trained for image classification is repurposed as the fully convolutional stream. A grid wise stream has been created in addition to the fully convolutional stream in our network. This stream embosses the saliency discontinuities along region borders, the visual contrast between super pixels, and collects semantic information about important items over multiple layers [23].

2 The performance of saliency detection in object perception is improved by using a multi-task FCNN with a dilated kernel-based technique, which performs a collaborative feature learning process. This also effectively enhances the feature-sharing capabilities of salient object recognition, resulting in a significant reduction in feature redundancy [24]. Using the proposed FCNN model's output, we present a fine-grained super-pixel-driven saliency refinement model.

3 Saliency models like the cutting-edge Amulet, UCG [25] are effective at detecting distracting regions since they typically stand out from their surroundings. This is extremely sensitive to local areas, such as the boundary of a salient item and the local distractions because low-level features are constantly reused. To tackle this issue, we propose a complementary network, called the distraction diagnosis network (DD-Net), which is used to identify distracting regions, lessen their negative effects, and eventually improve model performance [26]. The DD-Net learns to anticipate the distraction score (D-Score) from the input image and produces a noise-free output given to the next level of the model assuring the high-confidence regions. DD-Net decides which areas of a picture should be ignored while saliency detection is being done.

4 Further we build a sparse graph network, where every pixel in this network is connected to both its neighbors and the

node that shares a boundary with them has the greatest similarity to those neighbors. The proposed sparse network effectively utilizes the local spatial layout and eliminates unused nodes that distinguish one another one another.

5 Active contour is a technique used in computer vision to detect objects in an image. It is built on the concept of minimizing the energy of the segmentation [27]. The active contour method uses a curve that evolves to fit the boundaries of the object in the image. It does not subject to on the image gradient [28] and can detect objects even when the gradient does not consider object boundaries. The active contour method is used in conjunction with the convex hull technique to accurately detect the salient objects in an image.

To be more precise on the motivation and the contribution of the work, the following briefs it.

| Motivation | Contribution |
|---|---|
| 1) To Handle the Scale Variations in complex background: To investigate the methods for improving the robustness of salient object detection models to variations in object scales and ensuring consistent performance across diverse datasets and scenarios. | To attain the objective [1], VGG 16 architecture been completely remodified according to necessity of our work, especially three were three extra convolution layers (ECL) that are united with the top four pooling layers of the default network, this extra convolution layer performs the dilation operation to gain various scale information from different pooling layers and different dilation rates are used to confirm that no loss salient information occurs. |
| 2) To address Partial Visibility: To explore the techniques to enhance the ability of detection algorithms to localize salient objects accurately, even in the presence of partial visibility, low contrast etc. and improving overall detection performance. | The [2] objective is attained by developing a dense labeling component, which tries to find out the correlation between the two pixels. Equation (1) is formulated to perform the operation, most importantly identifying the appropriate threshold value through several trial been a challenging task and done successfully. |
| 3) To improve the Robustness to image distortions: To develop the techniques to enhance the robustness of detection models to common image distortions, such as blurring, noise, or compression artifacts; ensuring the reliable performance in various imaging conditions. | Objective [3] is one such uniqueness in our work, here the entire distraction detection network (DDNet) is designed for performing the distraction analysis. Formulated the functionality to compute the distraction score of the all the regions, using that the distraction mining is done at ease. |
| 4) To Integrate Semantic and conceptual Understanding: To develop an approach that combine visual saliency with semantic understanding to provide contextually relevant salient object detection, enabling more meaningful interpretations of detected objects within the image. | To have semantic and conceptual understanding of an image, it required to identify the superpixels in every region, and to find the relationship between the superpixels from different region. In our method every superpixel acts as node and the relationship between the nodes acts as an edges and the final graph construction forecast the semantic information that are present in an image. Formulated equation (4), (5) and (6). |
| 5) To optimize salient object detection through energy optimization: To develop a methodology by formulating energy functions that effectively quantify the saliency of objects within images, and incorporating iterative refinement processes to enhance segmentation accuracy by smooth object boundaries. | To obtain a smooth boundary of the salient objects, a network is designed, which computes internal energy and external energy amongst the saliency region to identify the boundary of the objects in an optimal way. Equation (7), (8) and (9) are formulated. |

We evaluate the suggested technique with ten state-of-the-art methods, with six benchmark datasets using a range of evaluation metrics. The results of the testing demonstrate that our

strategy performs better than the cutting-edge methods at the moment. The rest of the article is organized as follows.

Earlier research on the dilation algorithm, dense labeling, distraction mining, graph-based salient region detection, and active contour approach is reviewed in Section II. In Section III, we offer the salient region detection method we've recommended. With reference to the datasets for MSRA-B, ECSSD, HKU-IS, DUT-OMRON, PASCAL-S, SOD; which consists of train data, test data and validate data and those test data from the datasets were used for evaluation.

Section IV details the experimental design and results. In Section V, technical details of our framework are discussed, Section VI shows the quantitative comparison and visual comparison of our work with other state of art techniques and in Section VII we conclude the paper.

## 2 Related Work

The low-contrast image that we receive is to be pre-processed like contrast stretching to enhance the image's contrast. Fully Convolution Networks consist of multiple convolution layers where these layers apply a set of learnable filters to the input image capturing different features [29]. These filters can help in enhancing the edges and features for low-contrast images. Down-sampling layers like max pooling or stride convolutions help in network focus on capturing larger patterns in low-contrast images.

Skip connections are used in-between layers to preserve low-level features from earlier layers which are particularly valuable in the case of low-contrast images. Alongside Skip connections choosing activation functions is also as important as they will be helpful to introduce nonlinearity and improve the network's ability to capture the relationships between data. [30] uses parametric saliency and non-parametric clustering for saliency detection.

But low contrast images are easily misclassified so choosing an accurate and modified loss function is also important as it helps in semantic segmentation [31]. Different methods involved in extracting information in low-contrast image, histogram equalization, wavelet transform, image fusion, contrast stretching, and deep learning with the dilated kernel are a few.

### 2.1 Dilation Algorithm

Image segmentation grounded on deep learning has wide applications in autonomous driving, space exploration, the medical field, and in so many other fields [32]. Convolution neural network has a vital role in image segmentation and image classification, but at the same time, it takes too many resources for computation [33]. To address this problem [34] used CNN with a dilated kernel instead of using a convolution kernel, this reduced the training time and improved the accuracy of their model.

But in contrast, hybrid dilated CNN was built by stacking DCNN with different dilation rates on the kernel and still got further improvement in the accuracy of image classification. Deeplab framework [35] uses multi-scale neural networks using sparse convolution, their dilated network is used for feature extraction, and it focuses on high-level and low-level features to obtain highly accurate segmentation results.

The dilation filter [36] has a strong impact on edge detection algorithms to obtain accurate edge map results. The most difficult task of segmentation is performing it on low-contrast images, [37] uses morphological transformation for background approximation and has the conjunction of multiple background analyses.

In an application of classifying hyperspectral images [38] used MDCNN. This framework helps to reduce the complexity, efficiently concatenates every hyperspectral information in the map, and generates the feature map with a limited number of parameters.

Fig. 1 shows the reference diagram of using dilated kernel to classify an image, it mentions the necessity of using dilation kernel, in general using of Convolution neural network has a vital role in image segmentation and image classification, but at the same time, it takes too many resources for computation. To address this problem [39] used CNN with a dilated kernel instead of using a convolution kernel, this

reduced the training time and improved the accuracy of their model.

### 2.2 Dense Labeling

Dense labeling is a simple, faster, and efficient way of processing the saliency and its important concept is extracting knowledge from low-contrast images. [40] performs saliency estimation on macro objects and approximates the location of the object as well. In this regard [41] uses super pixel values to estimate the dense depth information in an image.

This framework has been designed with the idea of being generic in addressing all types of images. Dense labeling along with DCNN and SVM is formulated in for multiclass semantic segmentation. SVM explores different representations of multi-class objects which are indirectly related to identifying the target object and it is combined with DCNN for efficiently segmenting the respective class objects. Semantic segmentation is used not only for static images but also in addition for real-time images.

It employs an innovative approach to dealt with the images captured by multiple cameras, here the issue is information on the image is captured by only a few viewpoints, and the rest nullifies the output. So functional color-based labeling method is incorporated to attain the desired output [42].

For evaluation remote sensing image retrieval used the dense labeling concept, especially for the dataset that contains multi-classes. RSIR used features-based extraction with deep learning techniques for more efficient performance.

#### 2.2.1 Distraction Mining

In general, any segmentation process suffers from false positive or false negative values that distract the objective of the work [43]. So, the latest research on segmentation has a foresight on distraction identification, which focus on categorizing the noisy data that are merge with target objects and reduces the accuracy [44].

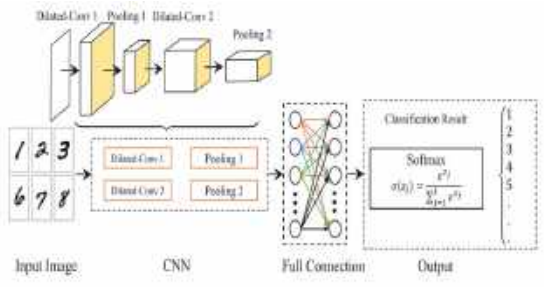



**Fig.1.** [30] CNN with dilated Kernel instead of using a convolution Kernel

The COS framework concentrated on the focus module to remove the distractive noisy background that is alike to the target object.

It estimates the distraction and its position for removal, through this the segmentation of the target object is made perfectly.

In medical image processing minimizing false ratio is one the most desirable outcomes, in spite of that DSNet [45] uses two models namely the neighbor fusion model to concatenate high-level features and the distraction separation model aims to catch object information by ignoring the distraction [46].

#### 2.2.2 Graph Construction

Graph-based segmentation has attained more popularity in recent years [47]. The graph-based approach divides the image as a graph of multiple subgraphs, each representing the expressive region of the image.

PSA [48] recognizes the indirect impact of background on the eloquent region through the graph to get accurate output. GBA generates a set of weights for the nodes and connectivity among the nodes to create the target image content.

Using Eigenvalues and covariance values [49] computes the weightage of the nodes for segmentation. The DWCut works efficiently with fewer number parameters also.

Fig. 2 describes about graph construction based on hierarchical clustering technique.

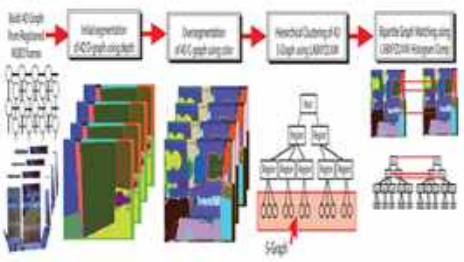


**Fig. 2.** [42] Graph based segmentation

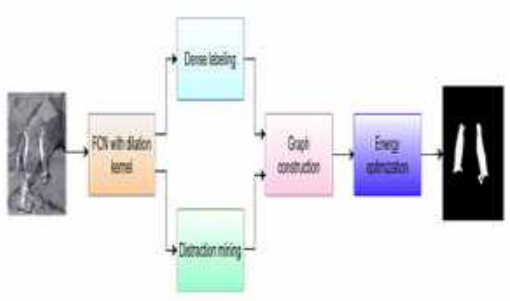



**Fig. 3.** Framework of GSD with distraction detection

## 3 Proposed Methodology

Fig.3 shows the framework of the proposed work, where the raw input image is passed over fully convolutional neural network for feature extraction and the dilation kernel with different kernel rate are convolved over the image to extract the feature with scale variance, saliency map along with probability value is generated as output and it is processed through dense labeling (DL) and distraction mining (DM) component.

DL segregates the foreground and background pixels based on the relationship between the pixels and DM is one which identifies the pixel region that were redirecting the detection process.

Further the output of DL and DM is processed and the graph is constructed based on the super pixel values and finally the energy optimization function brings out the smooth salient foreground image as an output [50].

**Pseudocode of our Framework**

1 Load the image and pass it through a fully convolutional neural network (FCN) for feature extraction.
2 Convolve the dilation kernel with different kernel rates over the image to extract features with scale variance and generate a saliency map along with probability values as output.
3 Process the saliency map output through dense labeling (DL) and distraction mining (DM) components.
4 DL: Segregate foreground and background pixels based on pixel relationships.
5 DM: Identify pixel regions that redirects the detection process.
6 Process the output of DL and DM, construct a graph based on the super pixel values.
7 Optimize energy function to obtain smooth salient foreground image as output.
8 Compute external and internal energy terms.
9 External energy: Combine data term and smoothness term using optimization technique.
10 Internal energy: Define constraints or penalties to enforce desired properties of the contour.
11 5.2. Obtain the saliency map.

### 3.1 Fully Convolutional Networks with Dilation Kernel (FCNDK)

FCN focus on understanding images at the pixel level and creating a comprehensive pixel-by-pixel regression network.

This network transforms the raw input image into a detailed saliency. By considering the vitality of contrast modeling and the size of objects in saliency detection, we consider the following issues into consideration when designing the structure of our FC network. On account to identify visual saliency, which always depends on low-level characteristics and high-level features for contrast modeling, the network should be significant enough to include multi-level information at the outset.

The FCN should then be able to project the object at different sizes, which is why scale-aware contrast modeling is also a priority. Lastly Training the images through every pixel is a necessary and extremely important task for accurate detection, for convenience, rather than starting the training for the model from scratch, we preferred to use a pre-existing network for simplicity.

One such kind of the most adopted and publicly accessible pre-trained model is VGG-16. However, for obtaining the output as per our requirement few modifications are incorporated on the default model of VGG-16, firstly the down sampling rate [51] is reduced by using the stride of 1 instead of 2, through which high-resolution saliency is obtained. Generally, FCN is accurate for making any prediction nevertheless, there are few limitations for FCN for segmentation, i.e., individual pixel contexts should be preserved for accurate pixel prediction.

It could be compensated by using dilation convolution in FCN. The advantage of using dilation convolution in FCN is, that it increases the attentive field of the convolution layer excluding extra parameters and as well which does not increase the computational complexity.

In FCN using pooling and strides sometimes reduces the spatial information of an image, whereas with the help of dilation convolution, more spatial pieces of information are preserved which is more vital for the segmentation process. Finally, by varying the dilation rate across the FCN, the model can adapt to gain knowledge on different patterns with different scales.

Therefore, using dilation convolution in FCN leads to improved performance of the model and is thus incorporated here. A framework like Caffee has built-in dilation convolution to efficiently control all these mentioned operations. Moreover, dilation convolution enables us to manage the density of the features in our designed FCN. Here is a comprehensive explanation of the working of VGG-16 in our model. VGG-16 have five max pooling layers (PL), which takes the input image and could able to create a large receptive field with more contextual information.

Additionally, it becomes complicated to process further rapidly and also increases the computational timing. So as a key three extra convolution layers (ECL) are united with the top four pooling layers, this extra convolution layer performs the dilation operation to gain various scale information from different pooling layers without losing any salient information.

The ECL down samples the pooling output to an equal-sized receptive field with the strides of 4,2,1,1 respectively. These four ECL outputs are unified with five PL outputs and a single output is generated, to which a sigmoid activation function is applied to acquire the probability of the saliency map.

### 3.2 Dense Labeling

Dense labeling is the process of separating the foreground object and background object in an image. The FCN has produced the salient probability map, which is further processed to segregate the foreground and background. Using the correlation heat map (CRHM) those pixels are processed to be represented in graphical form with different colors, which makes it simple to visualize the information and make the picture clear and accessible [52].

The idea is to take the saliency probability map values and plot color codes based on their values on the grid. The higher gradient (foreground region) is often represented with warmer colors and the lower gradients (background region) are represented with cooler colors.

The correlation coefficient is computed for the pair of probability maps. Two variances of correlation may arise, a positive correlation in which one feature tends to increase another one, and a negative correlation tends to decrease the other features. Heatmap (hmap) is generated based on the pair of values where positive represents the warmer color and negative uses the cooler color.

The equation for positive and negative correlation are as follows:

$$\mathrm{Cov}(\mathrm{m},\mathrm{n}) = \frac{\mathrm{E}[(\mathrm{m} - \mu\mathrm{Xm})(\mathrm{n} - \mu\mathrm{n})]}{\sigma\mathrm{m} * \sigma\mathrm{n},} \quad (1)$$

where,

Cov(m,n) represents the covariance between the two variables X and Y.

E denotes the expectation values of the terms that follows.

m - μXm & n - μn are the deviations of X and Y from their respective means μXm & μn.

μXm & μn are the means of X and Y respectively.

σm * σn are the standard deviations of X & Y respectively.

The resultant heatmap [53] is the graphical image of the correlation of the features, which helps in getting accurate segmentation in further progress.

### 3.3 Distraction Mining

Major issues with saliency detection are distracting the process with undesired regions, i.e. Which is not a region of interest (ROI). Distraction mining is a process used to identify distracting regions in an input image that can negatively affect the performance of a saliency detection model. The analysis is performed by a Distraction Detection Network (DD-Net), by predicting each pixel's probability value that is distracting. The DD-Net is trained using a distraction mining method, which simulates the absence of specific regions and evaluates the changes in saliency prediction.

Regions that cause negative prediction changes are considered distracting patterns and are used to train the D-Net. The analysis helps improve the performance of the saliency detection model by identifying and minimizing the negative effects of distracting regions. To lighten the unwanted sensibility and to attain better results we use a learnable network to detect the distraction that degrades the efficiency of the process.

The idea follows by mining the distraction from the input and then training the network to avoid the distraction. But in which way the mining is done, is the biggest problem; because doing more advanced calculations is not appropriate for mining the distraction from the input.

So, in our proposed work, we use the threshold values (TV) to accomplish the goal. All the actual values are in the form of binary images which represent saliency value as 1 and non-saliency as 0 and this is considered as a tweeting point.

As from the previous process ie. via FCN, a saliency probability map is already obtained, the probability map with a threshold value 0.6 and above is well thought of as a foreground region, and the region lesser than that is frozen as a background object, i.e., treated as a distraction. The forthright way of doing this is applying global thresholding, where the TV is chosen and for every saliency probability map, this threshold is applied and classified as foreground and background which is mentioned as a coarse grain saliency map (cmap). Equation:

$$\mathrm{P(i,j)} > \mathrm{Threshold.} \tag{2}$$

A coarse grain saliency map creates the saliency in a new dimension. The higher threshold pixel is determined in the previous step, through traversal of the salience pixel along with the four nearest pixels is performed with an elementwise product operator, and a new dimension is computed:

$$\mathrm{Cmap} = \mathrm{p1.p2.p3.p4}, \tag{3}$$

where, P(i,j) is pixel value with spatial coordinate i & j.

### 3.4 Graph Construction & Weighting

The hmap and cmap are diffused together to give the details of the feature in the image. These features represent the content of the image which includes intensity, and other low-level visual characteristics. The idea of a diffused saliency map is to highlight the salient regions and to suppress the background region.

Later the highlighted region is normalized to ensure the score lies in the region of 0 and 1 for better conception. In general, the spatial region of a salient object is lesser than the area of the background in an image [54], it motivates to the computation of the saliency seed for background and foreground separately and to construct the graph based on their similarities.

The complete image is split into two clusters, i.e., as foreground and background the process is further progressed. Clusters are denoted by a radius, the pixel at the centroid of the radius is picked as seed, and the surrounding pixels

around it is well thought of as neighbor pixels which are taken into consideration for similarity measure.

It compares the intensity of the center pixel with the neighbor pixel, if it outputs a higher value than or equal to the center pixel, that pixel is treated as a salient pixel, and a graph is constructed. The same procedure is followed for background cluster spatial areas also. The outcome of this process is going to be a super pixel feature vector of salient objects.

So far, a regular graph is constructed with super pixels and the weight of the super pixels is calculated for the saliency values of nodes. Weighing the super pixel is important because it becomes unreliable when the seed nodes are mixed with noises which it produces undesirable results and also fails to identify the salient objects. However, weighing the super pixels prominent the visuality. The further improvement of the salient feature is done by:

$$\mathrm{a(x,y)} = \mathrm{wxy}\,.\,\mathrm{dyy}, \tag{4}$$

$$\mathrm{b(x,y)} = \mathrm{a(x,y)} \,/\, \mathrm{k}, \tag{5}$$

$$\mathrm{S(x)} = \mathrm{b(x,y)}.\mathrm{S(j)}, \tag{6}$$

where,

S(x) is saliency value of foreground superpixel.

b(x,y) is normalized weight for a(x,y).

a(x,y) is weight of saliency value.

### 3.5 Energy Optimization

The last step is to attain the saliency confidence map and cut the edge of the boundaries of the object through the energy optimization (EO) concept. The working of EO completely relies upon the probabilistic value of the weighing pixel which was obtained using the graph-based weighing method and as a result, each pixel represents as if it belongs to the foreground or the background.

Usually, the traditional method works on the intensity cue but the EO generated the confident map of SO by contouring from the high probability values towards the boundaries and is treated as external energy of the saliency map. EO iteratively updates the shape of the saliency by moving laterally in a way that decreases the energy until it converges the object boundaries of the salient object.

The iteration process determines the optimal position that is incorporated in the objects. Meanwhile, the internal energy is also in charge of regulating the contour's regularity and smoothness from not differentiating out of the desired shape.

The values that are out of the external energy are taken as the background and not contoured. The optimal energy of saliency is given by:

$$\mathrm{EO} = \alpha * \mathrm{E_{internal}} + \beta * \mathrm{E_{external}}, \tag{7}$$

$$\mathrm{E_{external}} = \mathrm{Data_{term}} = \mid F_{ij} - S_{ij}, \mid, \tag{7.1}$$

$$\mathrm{E_{internal}} = \mathrm{Smoothness_{term}} = \parallel Centroid_{\,i} - Centroid_{\,j} \parallel, \tag{8}$$

where,

$\mathrm{E_{external}}$ is calculated upon the data term and $\mathrm{E_{internal}}$ is calculated using smoothness term.

data term represents the value of the super pixel feature Fij and Sij of the reference value and saliency value respectively at pixel(i,j).

$Centroidtroid_{\,i}$ and $Centroid_{\,j}$ represents the centroid of superpixels (i, j) respectively in smoothness term.

$\alpha$ and $\beta$ are the weighted parameters, by adjusting these parameters iteratively updates the saliency map.

Doing the above steps overcomes the problem of dealing with low-contrast images and also with complex images [55]. The iteration process plays a significant role in minimizing the total energy that an object possesses which leads to an accurate saliency map.

### 3.6 Salient Object Recognition - Saliency Map

Salient object recognition or saliency detection does not mean labeling the image, it focuses on grasping the intention of visually attentive objects, Saliency recognition deals with which part of the image has the salient object [56]. For labeling the detected object labeling methods can be used. The final output of the framework is a saliency map.

## 4 Experimental Results

The proposed work uses the six most widely used datasets [57] for evaluation purposes, the MSRA-B dataset used to facilitate the evaluation and development of algorithms for identifying salient regions in images.

It has 5000 images in the dataset that cover outdoor scenes, indoor scenes, natural objects, man-made objects, and many more along with ground-truth saliency maps which makes evaluation easier; moreover, due to the complexity and diversity of the images, it makes the model robust for saliency detection.

SOD [58] dataset has more than 5000 images with corresponding ground truth saliency maps. This dataset includes multiple objects with low contrast, which is again a challenging task for saliency detection.

Datasets from were referred, ECSSD, HKU-IS [58] which is from the previous work that contains complex scenes with challenging backgrounds and cluttered images. DUT from previous work, this dataset contains various levels of complexity and has 5168 challenging images, which have relatively complex and diversified contents. PASCAL from previous work and contains 850 natural images.

### 4.1 Evaluation Criteria

The evaluation metrics like Precision-Recall, F-measure and mean absolute error were used to assess the effectiveness of our model on salient object detection [59].

a) Precision provides positive prediction value ie. which says about true positive prediction value to how many were actually positive. The higher precision value indicates the model prediction positive outcome which is considered to be correct. Significantly false positive leads to crucial consequences:

$$Precision = \frac{|M \cap G|}{|M|}. \tag{9}$$

b) Recall speaks about the correct positive rate, i.e., the measure of the proposed work on the ability to properly identify all the positive instances. In detail it says about among all actual positive instances, how many the model predicted correctly. An increased recall value suggests that the model predicts all positive values correctly. Increasing the threshold increases the precision and decreases the recall rate and vice versa. It is so important to choose the appropriate threshold to determine the model effectiveness:

$$Recall = \frac{|M \cap G|}{|G|}. \tag{10}$$

c) F-measure [60] gives the harmonic mean of precision and recall, it considers both false positives and false negatives, which provide a single balanced value of both:

$$F_\beta = \frac{(1+\beta^2).precision.recall}{\beta^2.precision.recall}. \tag{11}$$

d) MAE stands for mean absolute error, which is the average error between the saliency map and the ground truth on a per-pixel basis. MAE provides a direct way to evaluate or to compare the model-predicted output with true values:

$$MAE = \frac{1}{H.W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left|S_{ij} - G_{ij}\right|. \tag{12}$$

## 5 Technical Details of Our Framework

For training the model with different datasets, GPU with RAM size of 32GB, along with 500GB HDD and PCIe 4.0 for high-speed SSDs is used. Linux operating system, Python 3.11 version with the framework like TensorFlow, PyTorch, Keras etc. been utilized.

### 5.1 FCNDK

For the training process all the images are resized to 320x320. The training is performed using the VGG-16 and is trained with a learning

**Table 1.** Quantitative evaluation results of GSD framework against ten state-of-the-art conventional saliency detection methods

| Datasets | Criterion | OURS | MFSS | DCL+ | DS | MDF | LO | SNET & DNET | GBMR | DRFI | GC |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **MSRA-B** | F-Measure | 0.942 | - | 0.931 | 0.898 | 0.885 | - | 0.923 | 0.8562 | 0.845 | 0.719 |
| | MAE | 0.034 | - | 0.042 | 0.067 | 0.066 | - | 0.045 | 0.1185 | 0.112 | 0.159 |
| **ECSSD** | F-Measure | 0.939 | 0.9341 | 0.925 | 0.9 | 0.832 | 0.8095 | 0.917 | 0.6617 | 0.782 | 0.597 |
| | MAE | 0.03 | 0.0316 | 0.058 | 0.079 | 0.105 | 0.1601 | 0.071 | 0.2318 | 0.17 | 0.233 |
| **HKU-IS** | F-Measure | 0.947 | 0.9305 | 0.913 | 0.866 | 0.861 | - | 0.922 | - | 0.776 | 0.588 |
| | MAE | 0.029 | 0.0302 | 0.041 | 0.079 | 0.076 | - | 0.059 | - | 0.167 | 0.211 |
| **DUTOMRON** | F-Measure | 0.917 | 0.872 | 0.811 | 0.773 | 0.694 | 0.7449 | 0.77 | 0.5631 | 0.664 | 0.495 |
| | MAE | 0.0382 | 0.0407 | 0.064 | 0.084 | 0.092 | 0.0758 | 0.118 | 0.1763 | 0.15 | 0.218 |
| **PASCALS** | F-Measure | 0.922 | 0.901 | 0.857 | 0.834 | 0.764 | 0.818 | 0.845 | - | 0.69 | 0.539 |
| | MAE | 0.048 | 0.0592 | 0.092 | 0.108 | 0.145 | 0.1695 | 0.103 | - | 0.21 | 0.266 |
| **SOD** | F-Measure | 0.873 | - | 0.848 | 0.829 | 0.785 | 0.7807 | 0.853 | - | 0.699 | 0.526 |
| | MAE | 0.072 | - | 0.12 | 0.127 | 0.155 | 0.1503 | 0.12 | - | 0.223 | 0.284 |

rate of 10-3. The weight decay [61] is provided as 0.0005, with momentum of 0.9 on training.

### 5.2 Dense Labeling

The dense labeling (DL) step in our work aims to predict a complete label mask to each of the pixel [62] of the input image. It uses a deep neural network (DNN) to conduct DL that directly outputs an initial saliency estimation for the given input image. The DL network is trained using a training dataset and it is designated to maximize the preservation of global image information and provide accurate location estimation of the salient object. The DL network architecture comprises of numerous layers including fully connected, pooling, and convolutional layers. The DL network takes the FCNDK output as input and produces a Heatmap as output.

### 5.3 Distraction Mining

The Distraction Detection [63] Network (DD-Net) is one of the components in proposed model for detecting salient objects. It is designed to identify and minimize the negative effect of distracting regions in an input image.

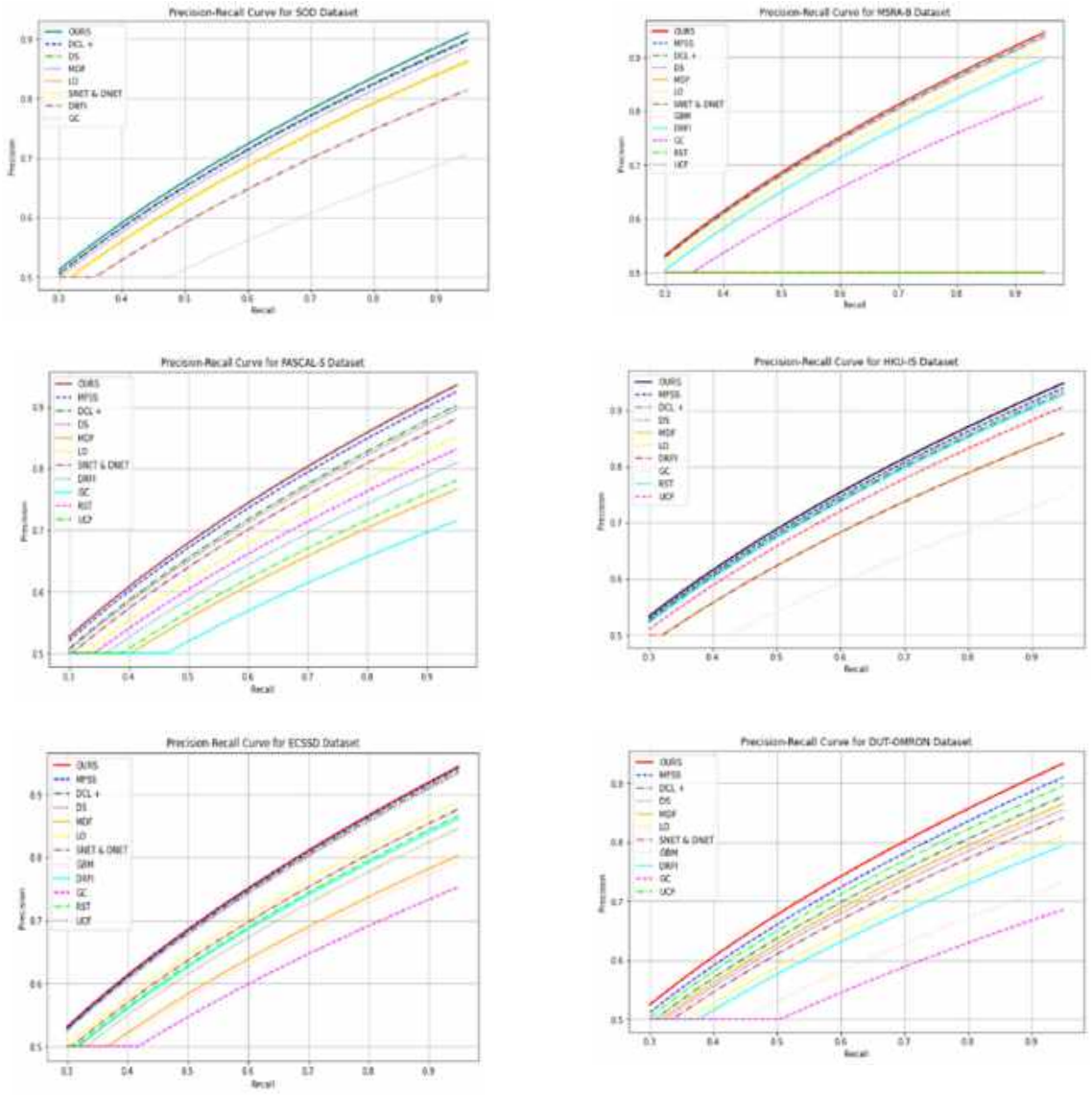



**Fig. 4.** PR curves of GSD against six state-of-the-art techniques saliency detection methods

The DD-Net learns to predict the distraction score for each location in the image. The distraction mining approach computes a distraction score for each location in the image using a sliding window.

The regions with high distraction scores are considered distracting and are masked out in the input image [64]. The modified image, with distracting regions removed, is then fed to the further process for saliency prediction. The DD-Net is trained with a mini-batch size of 10 for 20 epochs. The distraction mining is performed on a given image by computing the distraction score with a 20x20 pixel sliding window and stride with 2 pixels [65] are scanned through the entire image.

Overall, the DD-Net component reduces the negative influence and enhances the

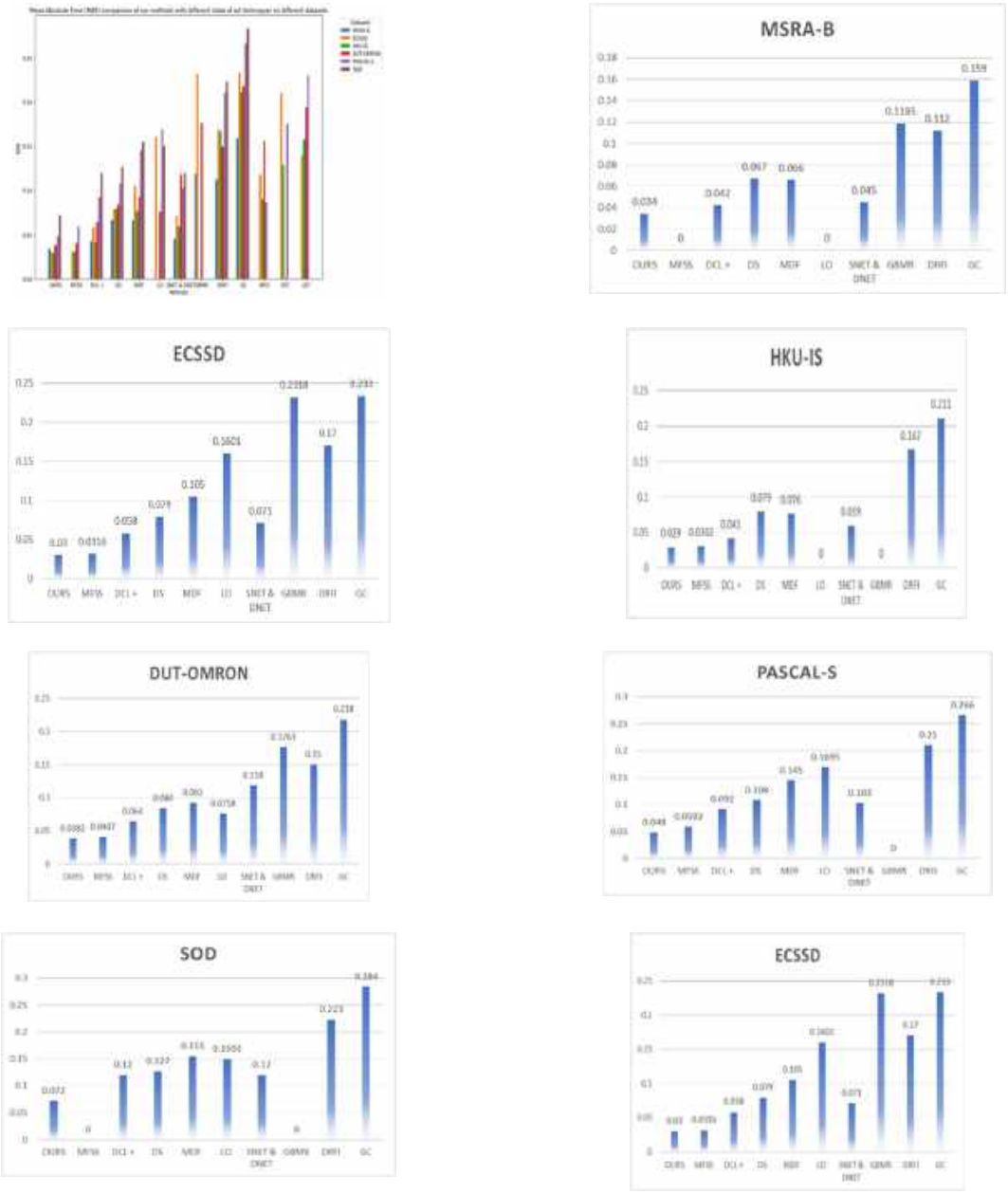



**Fig. 5.** MAE of GSD against six state-of-the-art techniques saliency detection methods

framework's performance by foreseeing the distracting regions on saliency prediction. It helps the model to concentrate on the salient object and produce better saliency predictions. It is implemented using the publicly available Caffe library. For the DD-Net, it is trained using a mini-batch size of 10 for 20 epochs, with a learning rate of 5×10-9. The outcome of the DD-Net is the distraction score for each location for the provided input.

This score indicates the likelihood of a particular region being distracting and negatively influencing the saliency prediction.

### 5.4 Graph Construction and Clustering

Generating the graph involves connecting each node to neighboring nodes [66] and the most similar node sharing a common boundary with the node next to it. Additionally, nodes on the four edges of the image are interconnected to reduce the geodesic distance between similar super pixels.

This sparse graph effectively uses the spatial relationship and removes dissimilar redundant nodes, ensuring better performance. The weight of each edge is determined based on the Euclidean distance between the mean Lab color features of the corresponding super pixels.

For constructing the graph, the proposed work encompassing three components: graph construction, robust sparse representation model with weights and saliency measure. An undirected regular graph with super pixels as nodes is created during the graph creation phase.

A local neighbourhood spatial consistency is taken into account when connecting each node to its nearby neighbours. The background seed nodes are chosen to be the image boundary nodes. The weight of the edges is distinctly created on the spatial distance between nodes. nodes on the four edges of the image are associated to each other.

This sparse graph effectively utilizes the regional spatial relationship and removes dissimilar redundant nodes. Overall, the graph construction for salient region detection involves connecting nodes based on spatial relationships and similarity, resulting in improved performance of saliency detection.

The resulting graph represents the super pixels and their relationships. The best representation coefficients are determined once the graph has been created and reconstruction errors are computed for each super pixel. This is done by solving an optimization model that minimizes the sparsity of the representation coefficients and the reconstruction errors. Iteratively updating both the reconstruction errors matrix and the representation coefficients matrix is the solution until convergence is reached.

### 5.5 Energy Optimization

In the proposed model, energy optimization has a vital role on detecting salient objects [67]. The energy values are regularized in the range of 0 and 1. The higher energy value of a pixel, the more likely it is to become a key point. The energy optimization process involves minimizing the energy function with respect to certain constants, which are expressed as functions of the active contour. The iterative model updates the active contour by the convex hull formed by the key points.

## 6 Output Comparison

### 6.1 Quantitative Details of the Proposed Method

The below table shows the performance of the proposed work. For fair comparisons with other models, we use the implementation results with recommended parameters and the saliency maps provided by the authors.

The quantitative evaluation result of our model along with existing cutting edges techniques are plotted as Precision Recall (PR) curve and MAE graph. The existing methods experimental values are used as such from the reference papers for the comparison.

From Fig.4 the PR curve of our GSD model performs well in terms of F-measure, GSD PR curve is higher than all comparison methods.

Fig.5. shows the mean absolute error is comparatively lesser than all other models. In the below graph MAE values is taken and compared our framework with other state of art techniques.

### 6.2 Visual Comparison

The visual comparison result shows the comparison of our framework with cutting-edge techniques. In the visual comparison it is clearly visible that our model performs better than the

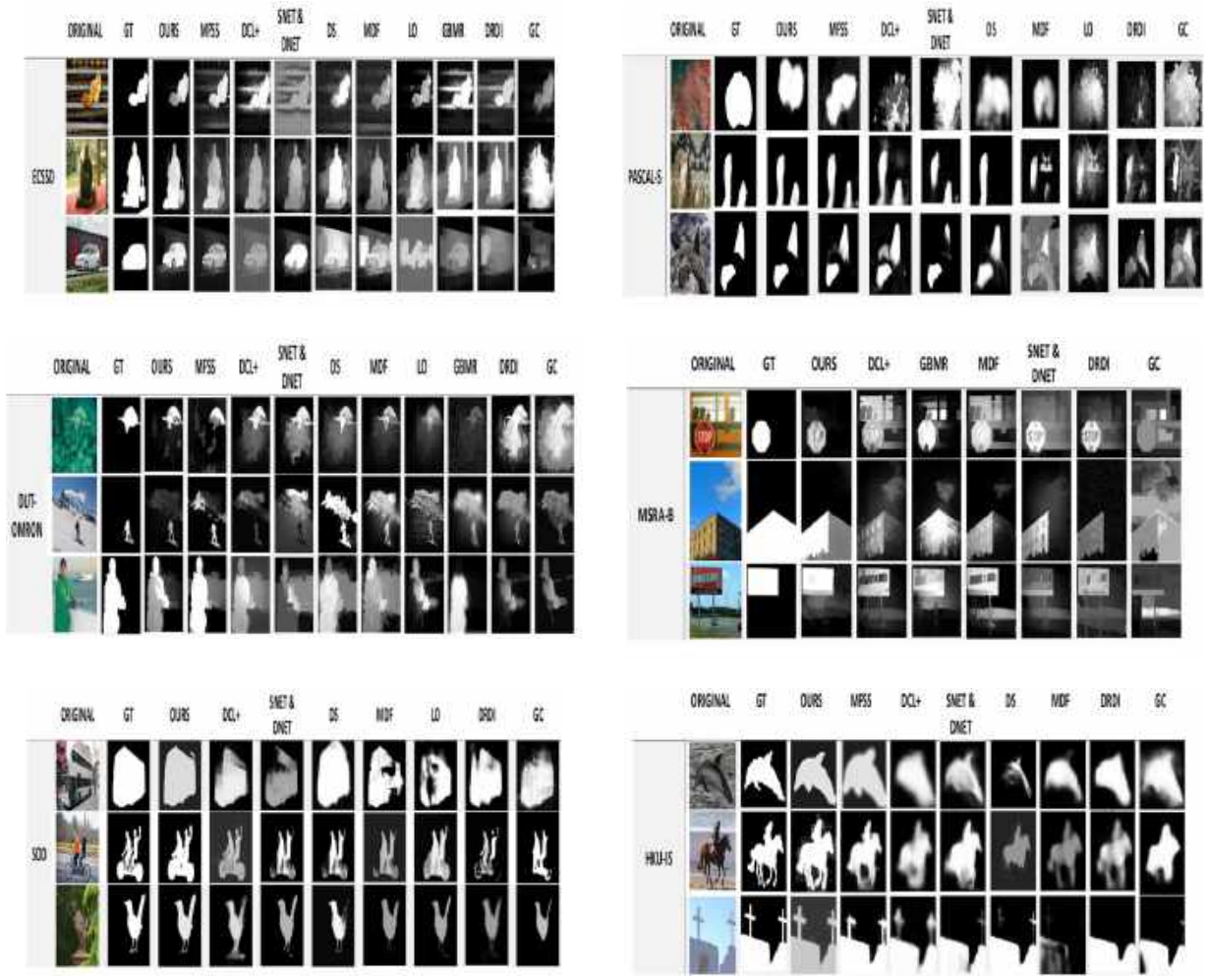



**Fig. 6.** Visual comparison of GSD framework with other cutting-edge techniques. Images with low contrast, with complex foreground and background patterns and highly interfering background are considered for comparison

existing models. To forecast the performance of our proposed model the images with low contrast, complex background and highly interfering background are considered for evaluation, and our model out performs for all the types of images while comparing with the existing state of art methods and the visual comparison proves that.

## 7 Conclusion

The suggested strategy is anticipated to perform better than current techniques in terms of accurately detecting salient objects, particularly in difficult situations with distractions and intricate backgrounds.

The accurateness and dynamicity of saliency detection are expected to be improved by the addition of multi-level features, dense labeling, distraction analysis, graph construction, and active contour refining.

In computer vision applications, the combination of these techniques is anticipated to increase the object recognition and scene comprehension accuracy and robustness of saliency detection. The efficiency of the

suggested strategy is confirmed by the experimental findings and comparisons with current approaches.

Moreover, in proposed work, the majority of the challenges that exists in salient object detection and recognition is addressed, nevertheless, the framework could further focus on handling the low-resolution images without involving too much complex functionality and to do the recognition with a mean time. This could be the one of the future works that can be augmented in our framework.

## References

1. **Li, X., Zhao, L., Wei, L., Yang, M. H., Wu, F., Zhuang, Y., Ling, H. Wang, J. (2016).** Deepsaliency: Multi-task deep neural network model for salient object detection. IEEE transactions on image processing, Vol. 25, No. 8, pp. 3919–3930. DOI: 10.1109/TIP.2016.2579306.
2. **Navalpakkam, V., Itti, L. (2006**). An integrated model of top-down and bottom-up attention for optimizing detection speed. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), Vol. 2, pp. 2049–2056. DOI: 10.1109/CVPR.2006.54.
3. **Liu, J. J., Hou, Q., Liu, Z. A., Cheng, M. M. (2022).** Poolnet+: Exploring the potential of pooling for salient object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 45, No. 1, pp.887–904. DOI: 10.1109/TPAMI.2021.3140168.
4. **Avidan, S., Shamir, A. (2023).** Seam carving for content-aware image resizing. Seminal Graphics Papers: Pushing the Boundaries, Vol. 2, pp. 609–617. DOI: 10.1145/3596711.3596776.
5. **Mejía, J., Ochoa-Zezzatti, A., Contreras-Masse, R., Rivera, G. (2020).** Intelligent system for the visual support of caloric intake of food in inhabitants of a smart city using a deep learning model. Applications of Hybrid Metaheuristic Algorithms for Image Processing, pp. 441–455. DOI: 1007/978-3-030-40977-7_19
6. **Jin, G. H., Zhou, Y. L., Yang, H., Hu, Y. T., Shi, Y., Li, L., Siddique, A. N., Liu, C. N., Zhu, A. D., Zhang, C. J., Li, D. Z. (2021).** Genetic innovations: Transposable element recruitment and de novo formation lead to the birth of orphan genes in the rice genome. Journal of Systematics and Evolution, Vol. 59, No. 2, pp. 341–351. DOI: 10.1111/jse.12548.
7. **Wang, L., Lu, H., Ruan, X., Yang, M. H. (2015).** Deep networks for saliency detection via local estimation and global search. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3183–3192. DOI: 10.1109/CVPR.2015.7298938
8. **Einhäuser, W., König, P. (2003).** Does luminance-contrast contribute to a saliency map for overt visual attention? European journal of neuroscience, Vol. 17, No. 5, pp. 1089–1097. DOI: 10.1046/j.1460-9568.2003.02508.x.
9. **Parkhurst, D., Law, K., Niebur, E. (2002).** Modeling the role of salience in the allocation of overt visual attention. Vision research, Vol. 2, No. 1, pp. 107–123. DOI: 10.1016/s0042-6989(01)00250-4.
10. **Cheng, M. M., Mitra, N. J., Huang, X., Torr, P. H., Hu, S. M. (2014).** Global contrast based salient region detection. IEEE transactions on pattern analysis and machine intelligence, Vol. 37, No. 3, pp. 569–582. DOI: 10.1109/TPAMI.2014.2345401.
11. **Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M. H. (2013).** Saliency detection via graph-based manifold ranking. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3166–3173. DOI: 10.1109/CVPR.2013.407.
12. **Wang, Q., Yuan, Y., Yan, P. (2012).** Visual saliency by selective contrast. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, No. 7, pp. 1150–1155. DOI: 10.1109/TCSVT.2012.2226528.
13. **Liu, D., Yamasaki, T., Wang, Y., Mase, K. Kato, J. (2022).** Toward extremely lightweight distracted driver recognition with distillation-based neural architecture search and knowledge transfer. IEEE Transactions on

Intelligent Transportation Systems, Vol. 24, No. 1, pp. 764–777. DOI: 10.1109/TITS.2022.3217342.

14. **Jiang, P., Ling, H., Yu, J., Peng, J. (2013).** Salient region detection by UFO: Uniqueness, focusness and objectness. Proceedings of the IEEE international conference on computer vision, pp. 1976–1983. DOI: 10.1109/ICCV.2013.248.
15. **Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H. Y. (2010).** Learning to detect a salient object. IEEE Transactions on Pattern analysis and machine intelligence, Vol. 33, No. 2, pp. 353–367. DOI: 10.1109/TPAMI.2010.70.
16. **Mai, L., Niu, Y., Liu, F. (2013).** Saliency aggregation: A data-driven approach. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1131–1138. DOI: 10.1109/CVPR.2013.150.
17. **Islam, M. A., Bruce, N., Wang, Y. (2016).** Dense image labeling using deep convolutional neural networks. 13th Conference on computer and robot vision (CRV), pp. 16–23. DOI: 10.1109/CRV.2016.31.
18. **Shu, C., Chen, Z., Chen, L., Ma, K., Wang, M., Ren, H. (2022).** Sidert: A real-time pure transformer architecture for single image depth estimation. arXiv preprint arXiv:2204.13892, pp. 1–7. DOI: 10.48550/arXiv.2204.13892.
19. **Paramanandam, K., Kanagavalli, R., (2022).** A review on deep learning techniques for saliency detection. Information and Communication Technology for Competitive Strategies (ICTCS 2021) ICT: Applications and Social Interfaces, pp. 279–289. DOI: 10.1007/978-981-19-0095-2_29.
20. **Jimenez, D., Cardoso-Moreno, M. A., Aguilar-Canto, F., Juarez-Gambino, O., Calvo, H. (2023).** PoSLemma: How traditional machine learning and linguistics preprocessing aid in machine generated text detection. Computación y Sistemas, Vol. 27, No. 4, pp. 921–928. DOI: 10.13053/CyS-27-4-4778.
21. **Sun, Y., Zhao, M., Hu, K., Fan, S. (2022).** Visual saliency prediction using multi-scale attention gated network. Multimedia Systems, Vol. 28, No. 1, pp.131–139. DOI: 10.1007/s00530-021-00796-4.
22. **Li, G., Yu, Y. (2018).** Contrast-oriented deep neural networks for salient object detection. IEEE transactions on neural networks and learning systems, Vol. 29, No. 12, pp. 6038–6051. DOI: 10.1109/TNNLS.2018.2817540.
23. **Jony, R. I., Woodley, A., Perrin, D. (2019).** Flood detection in social media images using visual features and metadata. 2019 Digital Image Computing: Techniques and Applications (DICTA), pp. 1–8. DOI: 10.1109/DICTA47822.2019.8946007.
24. **Zhou, L., Yang, Z., Zhou, Z., Hu, D. (2017).** Salient region detection using diffusion process on a two-layer sparse graph. IEEE Transactions on Image Processing, Vol. 26, No. 12, pp. 5882–5894. DOI: 10.1109/TIP.2017.2738839.
25. **Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B. (2017).** Learning uncertain convolutional features for accurate saliency detection. Proceedings of the IEEE International Conference on computer vision, pp. 212–221. DOI: 10.1109/ICCV.2017.32.
26. **Xiao, H., Feng, J., Wei, Y., Zhang, M., Yan, S. (2018).** Deep salient object detection with dense connections and distraction diagnosis. IEEE Transactions on Multimedia, Vol. 20, No. 12, pp. 3239–3251. DOI: 10.1109/TMM.2018.2830098.
27. **Yao, C., Kong, Y., Feng, L., Jin, B., Si, H. (2020).** Contour-aware recurrent cross constraint network for salient object detection. IEEE Access, Vol. 8, pp. 218739–218751. DOI: 10.1109/ACCESS.2020.3042203.
28. **Chen, E. H., Hu, H., Zeisler, J., Burschka, D. (2020).** Pixelwise traffic junction segmentation for urban scene understanding. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pp. 1–8. DOI: 10.1109/ITSC45102.2020.9294654.
29. **Bi, S., Li, G., Yu, Y. (2014).** Person re-identification using multiple experts with

random subspaces. Journal of Image and Graphics, No. 2, pp. 151–157. DOI: 10.127 20/joig.2.2.151-157.

30. **Chen, E. H., Hu, H., Zeisler, J., Burschka, D. (2020).** Pixelwise traffic junction segmentation for urban scene understanding. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pp. 1–8. DOI: 10.1109/ITSC45102.2020. 9294654.
31. **Zhao, R., Ouyang, W., Li, H., Wang, X. (2015).** Saliency detection by multi-context deep learning. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1265–1274. DOI: 10.1109/ CVPR.2015.7298731.
32. **Hernández-Herrera, P., Abonza, V., Sanchez-Contreras, J., Darszon, A., Guerrero, A., (2023).** Deep learning-based classification and segmentation of sperm head and flagellum for image-based flow cytometry. Computación y Sistemas, Vol. 27, No. 4, pp. 1133–1145. DOI: 10.13053/CyS-27-4-4772.
33. **Lei, X., Pan, H., Huang, X. (2019).** A dilated CNN model for image classification. IEEE Access, Vol. 7, pp. 124087–124095. DOI: 10.1109/ACCESS.2019.2927169.
34. **Liu, Y., Han, J., Zhang, Q., Wang, L., (2018).** Salient object detection via two-stage graphs. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 29, No. 4, pp. 1023–1037. DOI: 10.1109/TCSVT. 2018.2823769.
35. **Orhei, C., Bogdan, V., Bonchis, C., Vasiu, R. (2021).** Dilated filters for edge-detection algorithms. Applied Sciences, Vol. 11, No. 22, pp. 1–33. DOI: 10.3390/app112210716.
36. **Sreedhar, K., Panlal, B., (2012).** Enhancement of images using morphological transformation. arXiv preprint arXiv:1203.2514, pp. 33–50. DOI: 10.48550/ arXiv.1203.2514.
37. **Kumar, V., Singh, R. S., Dua, Y. (2022).** Morphologically dilated convolutional neural network for hyperspectral image classification. Signal Processing: Image Communication, Vol. 101, pp. 116549. DOI: 10.1016/j.image.2021.116549.
38. **Taleghani, N., Taghipour, F. (2021).** Diagnosis of COVID-19 for controlling the pandemic: A review of the state-of-the-art. Biosensors and Bioelectronics, Vol. 174, pp. 112830. DOI: 10.1016/j.bios.2020.112830.
39. **Yuan, Y., Huang, L., Guo, J., Zhang, C., Chen, X., Wang, J. (2018).** Ocnet: Object context network for scene parsing. arXiv preprint arXiv:1809.00916, pp. 1–22. DOI: 10.48550/arXiv.1809.00916.
40. **Cambra, A. B., Muñoz, A., Guerrero, J. J., Murillo, A. C. (2016).** Dense labeling with user interaction: an example for depth-of-field simulation. BMVC. DOI: 10.5244/C.30.102.
41. **Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M. M., Feng, J., Yan, S. (2016).** STC: A simple to complex framework for weakly-supervised semantic segmentation. IEEE transactions on pattern analysis and machine intelligence, Vol. 39, no. 11, pp. 2314–2320. DOI: 10.1109/TPAMI.2016. 2636150.
42. **Wu, M., Huang, D., Guo, Y., Wang, Y. (2020).** Distraction-aware feature learning for human attribute recognition via coarse-to-fine attention mechanism. Proceedings of the AAAI conference on artificial intelligence, Vol. 34, No. 07, pp. 12394–12401. DOI: 10.1609/ aaai.v34i07.6925.
43. **Lv, X., Persello, C., Huang, X., Ming, D., Stein, A., (2023).** DeepMerge: Deep learning-based region-merging for image segmentation. arXiv preprint arXiv:2305.19787, pp. 1–41. DOI: 10.48550/ arXiv.2305.19787
44. **Arevalo-Ancona, R. E., Cedillo-Hernandez, M., Ramirez-Rodriguez, A. E., Nakano-Miyatake, M., Perez-Meana, H. (2024).** Secure medical image authentication using zero-watermarking based on deep learning context encoder. Computación y Sistemas, Vol. 28, No. 1, pp. 199–210. DOI: 10.13053/ CyS-28-1-4898.
45. **Zhang, L., Liu, Y., Zhang, J. (2019).** Saliency detection based on self-adaptive multiple feature fusion for remote sensing

images. International journal of remote sensing, Vol. 40, No. 22, pp. 8270–8297. DOI: 10.1080/01431161.2019.1608384.

46. **Zhang, Y., Zhang, Y., Guo, W., Cai, X., Yuan, X. (2022).** Learning disentangled representation for multimodal cross-domain sentiment analysis. IEEE Transactions on Neural Networks and Learning Systems. DOI: 10.1109/TNNLS.2022.3147546.
47. **Isaac, S., Navon, R. (2013).** A graph-based model for the identification of the impact of design changes. Automation in Construction, Vol. 31, pp. 31–40. DOI: 10.1016/j.autcon. 2012.11.043.
48. **Bonfiglio, A., Cannici, M., Matteucci, M. (2023).** SoftCut: A fully differentiable relaxed graph cut approach for deep learning image segmentation. International Conference on Machine Learning, Optimization, and Data Science, pp. 497–511. DOI: 10.1007/978-3-031-53969-5_37.
49. **Lu, S., Mahadevan, V., Vasconcelos, N. (2014).** Learning optimal seeds for diffusion-based salient object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2790–2797. DOI: 10.1109/CVPR.2014.357.
50. **Wang, H., Jiang, L., Zhao, Q., Li, H., Yan, K., Yang, Y., Li, S., Zhang, Y., Qiao, L., Fu, C. Yin, H. (2021).** Progressive structure network-based multiscale feature fusion for object detection in real-time application. Engineering Applications of Artificial Intelligence, Vol. 106, p. 104486. DOI: 10.10 16/j.engappai.2021.104486.
51. **Shayegh, P. Ghanavati, S. (2017).** Toward an approach to privacy notices in IoT. IEEE 25th International Requirements Engineering Conference Workshops (REW), pp. 104–110. DOI: 10.1109/REW.2017.77.
52. **Nguyen, D. T., Ogunbona, P. O., Li, W. (2013).** A novel shape-based non-redundant local binary pattern descriptor for object detection. Pattern recognition, Vol. 46, No. 5, pp. 1485–1500. DOI: 10.1016/j.patcog.20 12.10.024.
53. **Wu, H., Li, G., Luo, X. (2014).** Weighted attentional blocks for probabilistic object tracking. The Visual Computer, 30, pp. 229–243. DOI: 10.1007/s00371-013-0823-3.
54. **Wang, Z., Chen, T., Li, G., Xu, R., Lin, L. (2017).** Multi-label image recognition by recurrently discovering attentional regions. Proceedings of IEEE Conference ICCV, pp. 464–472. DOI: 10.48550/arXiv.1711.02816.
55. **Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., Jagersand, M. (2019).** Basnet: Boundary-aware salient object detection. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 7479–7489. DOI: 10.1109/CVPR.20 19.00766.
56. **Contreras-Cruz, M. A., Martinez-Rodriguez, D. E., Hernandez-Belmonte, U. H., Ayala-Ramirez, V. (2019)**. A genetic programming framework in the automatic design of combination models for salient object detection. Genetic Programming and Evolvable Machines, Vol. 20, No. 3, pp. 285–325. DOI: 10.1007/s10710-019-09345-5.
57. **Yuan, Y., Fang, J., Lu, X., Feng, Y. (2019).** Spatial structure preserving feature pyramid network for semantic image segmentation. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), Vol. 15, No. 3, pp. 1–19. DOI: 10.1145/ 3321512.
58. **Yuan, Y., Li, C., Kim, J., Cai, W., Feng, D. D. (2016).** Dense and sparse labeling with multidimensional features for saliency detection. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 28, No. 5, pp. 1130–1143. DOI: 10.1109/TCSVT.20 16.2646720.
59. **Xiao, F., Deng, W., Peng, L., Cao, C., Hu, K., Gao, X. (2018).** Multi-scale deep neural network for salient object detection. IET Image Processing, Vol. 12, No. 11, pp. 2036–2041. DOI: 10.1049/iet-ipr.2018.5631.
60. **Cuevas-Olvera, M. A., Reyes-Reyes, R., Ponomaryov, V., Cruz-Ramos, C., (2018).** Salient object detection in digital images based on superpixels and intrinsic features. IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT), pp. 583–588. DOI: 10.1109/DESSERT.2018.8409199.

**61. Wang, H. J., Chen, L. W., Lee, H. Y., Chung, Y. J., Lin, Y. T., Lee, Y. C., Chen, Y. C., Chen, C. M., Lin, M. W. (2022).** Automated 3D segmentation of the aorta and pulmonary artery on non-contrast-enhanced chest computed tomography images in lung cancer patients. Diagnostics, Vol. 12, No. 4, pp. 1–16. DOI: 10.3390/diagnostics 12040967.

**62. Zhou, W., Newsam, S., Li, C., Shao, Z. (2018).** PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. ISPRS journal of photogrammetry and remote sensing, Vol. 145, pp. 197–209. DOI: 10.1016/j.isprsjprs. 2018.01.004.

**63. Fuentealba, P., Illanes, A., Ortmeier, F. (2019).** Cardiotocographic signal feature extraction through CEEMDAN and time-varying autoregressive spectral-based analysis for fetal welfare assessment. IEEE Access, Vol. 7, pp. 159754–159772. DOI: 10.1109/ACCESS.2019.2950798.

**64. Singh, M., Govil, M. C., Pilli, E. S. (2018).** CHACT: Convex hull enabled active contour technique for salient object detection. IEEE Access, Vol. 6, pp. 22441–22451. DOI: 10.11 09/ACCESS.2018.2826924.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is P Kokila.*

ISSN 2007-9737

# DDoS Attacks in Traffic Flow Streams Using Ensemble Classifiers

Dutta Sai Eswari*, Panga V. Lakshmi

GITAM University, Department of CSE,
India

saieswari3@gmail.com,vpanga@gitam.edu

**Abstract.** The failure of internet networking systems, which can happen in various methods, may negatively impact contemporary information and communication technologies. In these circumstances, DDoS attacks have targeted a growing number of organizations. These attacks use a deluge of demands for computation and communication resources to order a service unavailable to genuine users. Distributed denial-of-service (DDoS) attacks must be prevented on vital resources. This manuscript's most important new development is the DDoS attack defence ensemble classifier model. The suggested model uses these specifications to enable a drift detection feature and includes defining streaming properties for service requests. Additionally, an ensemble classifier is used to detect changes in the pattern of service request traffic. Statistical metrics like true negative rate, positive predictive value, and accuracy were used to test and analyze the service request stream synthesis results. Comparing the model to other benchmark models discussed in recent academic works would have been another tactic that could have been used to increase the model's importance.

**Keywords.** Distributed denial of service (DDoS) attacks, ensemble classifier, drift detection.

## 1 Introduction

The way news spreads has significantly changed as a result of developments in Technology (ICT). The way people receive and share information has significantly changed due to the information flow through the internet. The services supported by ICT are improving information exchange and continuum operations. Information and communication technology (ICT) system issues have a negative effect on many facets of contemporary society and the digital world.

The failure of these networks can be attributed to several things, including viruses and unauthorized users. DDoS attacks, or distributed denial-of-service attacks, involve the coordinated use of numerous computers and networks to deny access to authorized users. Over the previous ten years, it has constantly been a threat.

Therefore, these evil attempts to seize vital resources must be stopped. These DDoS assaults demand an ensemble-based classifier with minimal computational requirements and high decision-making accuracy in ambiguous data. The scholar suggests a method known as distributed denial of service attacks to model the distribution statistics of network flows, identify unusual attacks, and meet the network's requirements.

Numerous new developments and considerable challenges exist as the system network expands exponentially and the internet networks become more complex. The behaviour of network attacks significantly increases the difficulty of discovery. IT security experts are growing concerned about distributed request floods, which are used to deny service to one or more target servers. Numerous reasons could be responsible for such floods.

Distributed denial-of-service (DDoS) distribution, which has a global scope, can mitigate some elements of distributed denial-of-service (DDoS) attacks but not all of them. For example, telco clouds and networks need to be protected at a more granular level, but this is not feasible. This is because specific characteristics call for targeted, community-based intervention.

Signature-based detection and mitigation techniques are frequently used to thwart DDoS assaults. This is accurate to a very large degree. Their independence in understanding the

distinctions between regular and attack traffic patterns is crucial to their success on the battlefield. Despite their inventiveness in defending against typical attack types, this is where they excel. The accuracy of conventional detection models for detecting DDoS patterns has dropped due to the emergence of new security threats.

Targeted zombie armies that are remotely controlled can help mitigate distributed denial of service (DDoS) assaults. When cybercriminals initiate an attack of this nature, they fully exploit the compromised system by stealing its data or utilizing all of its memory and bandwidth. Because distributed denial of service assaults only utilize a small portion of the available computational resources, they have been proven ineffective and expensive [1].

DDoS attacks, also known as distributed denial of service attacks, are increasing across all kinds of networks, including those used for cloud computing, according to recently gathered data. On modern networks, distributed denial of service attacks [2] appear in variousflavours [3]. RUDY and control packet overflow are only two potential attack types. (R-U-Dead-Yet) With RUDY, attacking a target server is as easy as flooding it with many infinitely lengthy sessions.

When this occurs, there is no longer any responsiveness to fresh queries from other users. A more recent attack called HTTP POST/GET [4] sends legally posted messages to a web server that hosts an application very slowly. The web service will experience prolonged downtime and possibly even failure if subjected to a distributed denial of service attack.

This kind of assault, which can come from numerous other vectors, is modelled after the SQL injection attack paradigm (SIDDoS). The model [5] then tries to access database resources the opponent cannot access. This research discovered that the mean square distance for achieved features decreased as the number of sessions in the network traffic corpus increased. Given the exponential increase in the network traffic corpus, this outcome shouldn't be unexpected.

Because the data are probabilistic rather than deterministic, features determined to be insignificant are a consequence of this.

The rest of the article is prearranged as follows: section 2 discusses the related work, section 3 proposes a defence of DDoS attack in traffic flow streams, section 4 deliberates the experimental outcomes, and section 5 concludes the research paper.

## 2 Related Work

There are two main categories for DDoS intrusion monitoring systems: analysis of the data source being transferred and traffic analysis. The most common detection and mitigation models and their benefits and drawbacks will be discussed below.

### 2.1 Analysis of the Transmitted Data Source

The preliminary investigation of IDS is the main topic of this part of [6], which also chronologically lists important DoS. The attacks SYN-Flood, HTTP-flood, and Smurf are just a few that have significantly altered the timeline. Most of this study falls into one of three categories: anomaly-based tactics, signature-based techniques, or a combination of both [7].

Neural networks have been created and usedto spot possible real-time distributed denial of service attacks. Researchers discovered layer seven DDoS flooding attacks were extremely potent in frequency and quantum in the study mentioned in [8].

These models' intrusion detection accuracy rates, however, are generally insufficient. Only the NB classifier can learn and build with a significant precision rate while operating at considerably faster speeds than other benchmark approaches [9, 10] studied the dynamic interaction between malicious traffic patterns' temporal and geographic correlation.

This model detected malicious traffic flows without needing router-level adjustments to IP forwarding schemes. Researchers in [11] used ensemble NNs composed of RBF and SOMs to identify malicious traffic and achieved remarkably high accuracy rates.

The success of the research was largely due to the use of an ensemble of NNs. The referenced research [12] claims that the accuracy of the multilayer perception NN was dependent on

neurons present in the omitted layers. Results from the study's trials revealed accuracy rates above 90%, which is regarded as sufficient in the industry. The authors of [13] employed a bi-layer feed-forward neural network to distinguish between fake and genuine attacks. The KDD Cup99 corpus simulation results showed the model to have substantial performance and respectable levels of precision.

In the HIDE model [14], NNs are used in addition to pre-processed statistical values to classify the movements. In this research, various classification techniques are compared and examined. These approaches consist of BP, RBF, Fuzzy-ARTMAP, and PBH. The PBH and BP models demonstrated noticeably higher levels of detection accuracy when compared to the outcomes produced by the overwhelming majority of the other models. [15] Traffic flow classification is carried out using a method based on detecting anomalies.

A 96% RBF neural network-based classifier detection rate was determined empirically using the UCLA dataset. Fuzzy clustering methods, artificial neural networks, and evolutionary algorithms have all been used to study the difficulty of detecting Distributed Denial of Service assaults.

The study mentioned above [16] created an original model incorporating multilayer perceptions, back propagation, and SVMs. This model was put forth as a solution to the problem and includes sub-models like packet gathering and data pre-processing. A 90.78 percent success percentage in experiments was noted. An overview of both a conventional DoS assault and a distributed DoS attack that could be carried out in the cloud is provided by the authors of the article [17]. Any attack has the potential to disrupt things significantly.

They also suggest defence mechanisms, safety measures, and technical safeguards. Additionally, they discussed a wide range of difficulties and problems associated with preventing DoS attacks in a cloud setting. This highlights the gravity of the issue raised by DDoS attacks on cloud computing, the range of solutions that can be used to address it, and the potential for additional study into mitigating and preventing these attacks.

A formal paper [18] establishes the notion of security known as "non-malleability" under selective opening attacks (NM-SO security). They also examine the connections between the different NMSO security concepts and those between NMSO security and conventional SOA security, as well as non-malleability and conventional security concepts.

Refer to reference [19] to learn more about the spread denial of service attack carried out using a botnet against the application layer of the web server. There are descriptions of events from different parts of the world and financial losses from well-known businesses and official websites; this indicates that extra caution should be used and that more investigation should be done to evaluate the situation fully. [20] suggested an efficient and secure algorithm for solving large systems of linear equations, which is essential and time-consuming across many engineering fields.

The proposed approach can manage non-singular dense matrices, making it sufficiently general to be applied to even the most complex adversary models (a.k.a. fully malicious model). The algorithm used here is more effective and accurate, and the used approach is the current industry gold standard in terms of checking capabilities.

This model type usually ignores the heterogeneity of the features trained into the database. Additionally, malicious traffic flows hint at the consistency of the results obtained for the variables under consideration. The findings indicate drawbacks to expanding the training pool and lowering the approaches' accuracy.

### 2.2 Analysis of Traffic Flow

An anomaly-based methodology for analyzing traffic flows is proposed in the article [21]. The article's authors predicted [22] that the model may identify DDoS attacks by excluding the offending nodes if traffic flows were clustered according to their IP source addresses. They believed that this was the situation. The study's strength rests in its ability to gather numerous unique sets of attributes despite the high false-positive rate that was seen.

When compared to other recent works on the same topic, this made the study stick out as being very effective. The study in [23] focused heavily on multi-layered artificial neural networks (ANNs) to identify and thwart DDoS attacks. As an additional

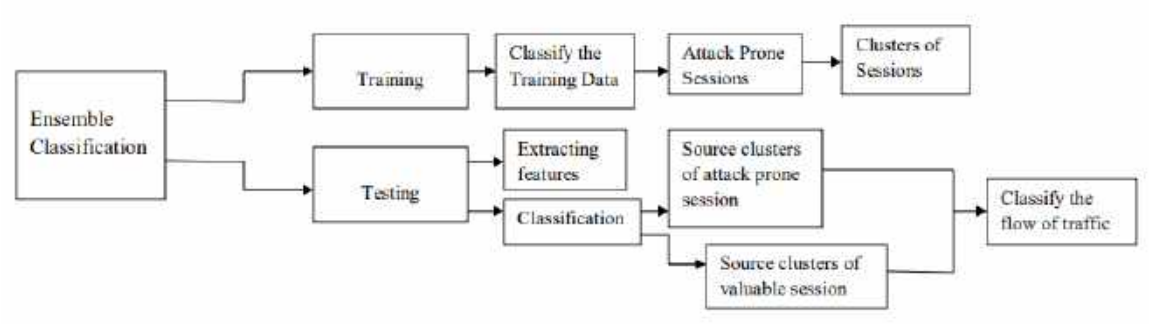



**Fig. 1.** Operational flow of the proposed model3.1 KS-Test

bonus, traffic flows have been correctly classified as legitimate or malicious using machine learning in combination with NB's classification algorithm [24].

According to a recent study, many academics and researchers praise the value of ensemble learning methods for traffic flow analysis. In the works discussed in [25–27], multi-labeled network flows that reflect various types of real-world traffic are categorized. These compositions rank among the most significant of all time.

Furthermore, the models incorporate a predictor at the kernel level. The simulations showed that the Kernel-based approach [28] offered the highest accuracy when tested on the NSL-KDD corpus. [29] incorporates an adaptive neuro-fuzzy inference method to differentiate between normal and vulnerable traffic flows.

The technique was used to train various models, each with its own data collection. The findings from each classifier were combined in the final phase after each had undergone its own set of tests. The model depended on several static corpora, and the evaluation results carried out using dynamic, practical test corpora were unacceptable, so it has been determined that the experimental study's findings are meaningless. As was already mentioned, all of the versions share this flaw.

The traffic flow characteristics on the network are not accurately reflected by the features used to train classifiers. The bio-inspired flood IDS was trained using the new method, which included adding a new attribute set to enhance this method. The Cuckoo Search algorithm was the foundation for its creation due to its binary classification capabilities. In the empirical analysis of the method, a precision rate of 95+% and a FAR of 4.5+% were discovered.

However, only a corpus of 7100 sessions, all of which consisted of identical transactions, were used in the simulation study. A review of recent studies shows that the success of machine learning techniques depends on selecting the appropriate features and applying the appropriate values during the learning phase. Additionally, the number of datasets increases with the availability of more information.

There are also high FAR rates in traffic analysis methods. This article attempts to categorize particular training corpora into categories based on the diversity of the distribution of their contents.

All classification models taken into account as anensemble component are additionally trained using information from the relevant cluster sets. The suggested solution stands out compared to other contemporary models due to its distinctive application of the metrics and standard learning methodology used in contemporary attribute selection.

As the previous paragraphs show, modern models heavily depend on static databases to determine detection rates. However, this approach uses dynamic databases that combine actual and simulated network attacks. The final step is to select one of these dynamic data repositories. Based on the survey, there are several issues with existing methods in attaining high attack prediction and accuracy.

Hence, this study proposes the ensemble classification model for DDoS attack detection in traffic flow streams.

# 3 Defence of DDoS Attack in Traffic Flow Streams

## 3.1 KS-Test

In supervised learning, each input is linked to an output label, allowing the system to learn from labelled data. Ensemble classifiers enhance the model's overall accuracy and resilience by combining the predictions of several basic classifiers.

Combining several classifiers, an ensemble classifier aims to minimize the error (misclassification rate) caused by an inadequate classifier. The fundamental premise is to get the predictions of many classifiers using the initial data and then merge their predictions to form a robust classifier.

Figure 1 shows the operational flow of the proposed model. Devices or sensors connected to the network gather data on traffic flow. These data sets often include information regarding packet the aders, including source and destination IP addresses, port numbers, protocol type, and timestamps.

Several relevant features are collected from the data on the flow of traffic. These features may include but are not limited to packet rates, packet sizes, inter-arrival periods, protocol distribution, and other features.

The labels applied to the data indicate whether each network flow indicates normal activity or a DDoS attack. This tagging may be implemented manually or automatically by making use of attack patterns that are already known. An ensemble classifier, which is made up of numerous ensemble classifiers, is trained with the help of the labelled data.

Each ensemble classifier can make use of a different algorithm or feature ensemble. The ensemble classifier uses methods such as majority voting and weighted averaging to aggregate the predictions made by its basis classifiers.

To assess the effectiveness of the trained ensemble classifier in identifying distributed denial of service attacks, it is tested on new traffic flow data that it has not seen before. Attributes like accuracy, precision, recall, F1-score, and ROC curves are used to assess the ensemble classifier's effectiveness. Because of this, its ability to differentiate between regular traffic and distributed denial of service assaults can be evaluated more accurately. The study [30] and potential future research may use single or multiple bio-inspired algorithms to find anomaly-based diverging flood detection.

The strategy outlined in this chapter also has the potential to be improved further to recognize different types of attacks that might be met in networks, like flash crowds. The KS-Test is used to assess the mathematical similarity of two vector distributions. The formulation of this assessment is as follows: Incorporate the sum of v1and $v_2$'s vector values $\|v_1\|$ , $\|v_2\|$ into your calculations. A sequence of given vectors can be used with the following equation 1 to calculate the cumulative ratio of its entries:

$$\begin{cases} cr = 0 \\ \bigvee_{j=1}^{|v_i|} \{e_j \exists e_j \in v_i\} \, begin \\ \quad cr = \frac{e_j}{\|v_i\|} + cr \\ \quad CR_{v_i} \leftarrow cr \\ \quad end. \end{cases} \tag{1}$$

The procedure is performed many times overall because each supplied vector vi has its cumulative ratio elements computed several times. Calculates the ratio of the element ej to the weighted average of the goal vector $v_i$.

The sequence of an element that occurs in the supplied vector $v_i$ as element $e_j$ is used to determine the iteration process for each vector. The cumulative ratio cr is then applied to the sum of these values.

The cumulative ratios of every element in the input vector are then added to create a collection of numbers known as CRvi. The $CR_{V_1}$ $CR_{V_2}$ sets are tested and saved to identify cumulative ratios, and the ratios of the components present in the target vectors $v_1$, $v_2$ are analyzed.

The KS-test then reveals the exact distance between the two vectors' cumulative ratios, indicated by a similar number in the two sets $CR_{V_1}$ , $CR_{V_2}$, and preserved in the set $diff(CR_{V_1} \leftrightarrow CR_{V_2})$. This is demonstrated in Equation 2:

**Table 1.** How the algorithm's code is employed

| Code | Representation |
|---|---|
| $SL=\{s_1,s_2,\ldots s_{\|SL\|}\}$ | Based on the session starting time, the list is sorted in order. |
| $TSL \leftarrow SL$ | Replicate the sorted sessions list. |
| $b\ (s_i)$ | the session $s_i$start time |
| $e\ (s_i)$ | the session $s_i$ending time |
| $scl_j$ | the forming of the j[th] cluster, with j = 1 to start |

$$diff\left(CR_{V_1} \leftrightarrow CR_{V_2}\right) \leftarrow abs\left(cr_i - cr_j\right). \quad (2)$$

The maximum number of the d-stat, or distance measuring statistic $diff\left(CR_{V_1} \leftrightarrow CR_{V_2}\right)$, is also displayed. The combined values of $\nu_1$ and $\nu_2$ for the target vectors-critic at the probability degree VIZ threshold are then displayed using the KStable; depending on the TPR distribution, common values for this threshold are 0.01, 0.05, and 0.1. The distributions of the vectors $\nu_1$, $\nu_2$ are either similar or different from one another, respectively, when the d-stat is greater than the d- critic.

### 3.2 Clustering the Sessions

The target traffic flow sessions are organized coherently in this article. We will also examine how each possibly vulnerable and desirable subset of sessions is handled separately. We organized the sessions from the earliest to the latest start time based on label traffic patterns and the total number of sessions.

The process entails sublists of sessions being grouped per a predetermined scheme, carried out repeatedly until the complete session list is in chronological order.

The session that has been enrolled in the cluster in ascending order of the listed session is used as the cluster centroid scli after the cluster formation process has been repeated several times for each cluster.

Sessions are sent to the cluster sclito sort when their start times coincide with the centroid's end time. Select the scli cluster centroid with the most sessions with a " Concluded " time stamp within the target scli cluster.

When the centres of each scli cluster remain constant throughout time, this verifies the specified order of cluster creation.

If the newly chosen centroid is not part of the same cluster as the one that was previously used, the sessions are rearranged so that their start time is earlier than their end time relative to the centroid of the cluster scli. The method involves iteratively selecting a new centroid and moving sessions to the target cluster until the centroid is secure. The idea behind the sessions clustering algorithm is depicted in Table 1.

### 3.3 Usage of Characteristics

The distributional similarity of the following inputs indicates how closely the session groups are related. The following distribution similarity data can be used to demonstrate the session cluster consistency.

1 A vector whose length is proportional to the size of each cluster is used to symbolize the initial intervals of a session. All groups keep tabs on the overall session count. How the cluster graphically depicts the session start time vector:

$$\forall_{j=1}^{|C_i|-1} \left\{ sbi\left(C_i\right) \leftarrow \left(bt(s_{j+1}) - bt(s_j)\right)\right\} \quad (3)$$

Intended total sessions present in cluster $C_i$, in the sequence given $bt(s_j)$, $bt(s_{j+1})$, take the difference in seconds between session $s_j$ and $s_{j+1}$and append it to vector $sbi\ (C_i)$.

**Algorithm 1:** Sessions grouping algorithm

Loop 1: *while*($|TSL| > 0$) Begin // whereas $TSL$ is not vacant

$c_j = \{s_1 \exists s_1 \in TSL\}$ // take the orderd epicted the first session in list $TSL$ as centroid of the $j^{th}$ cluster $scl_j$

$scl_j \leftarrow c_j$

$nc_j \leftarrow c_j$

Loop 2:

$\underset{i=1}{\overset{|TSL|}{\forall}} \{s_i \exists s_i \in TSL \wedge s_i \neq c_j\}$ begin

*if* $(b(s_i) < e(c_j))$ begin

$scl_j \leftarrow s_i$

*if* $(e(s_i) > e(c_j))$ begin

$nc_j \leftarrow s_i$

End

End

End

*if* $(c_j \neq nc_j)$ begin

$c_j = nc_j$ // the cluster $scl_j$ new centroid

$scl_j = null$ // the cluster becomes nil

Go to loop 2

End

*else if* $(c_j \equiv nc_j)$ begin

$TSL = \{TSL \setminus scl_j\}$ // the cluster $scl_j$ entries are detached from the sorted list $TSL$

$j = j + 1$

End

End

2 Each cluster's completion time is represented by a vector $sbi\ (C_i)$ of size $|C_i| - 1$, and this vector represents the session's Completion Intervals. The number of sessions in each cluster is represented by the symbol $|C_i|$. In this example, the session intervals for cluster $C_i$ are represented by the vector $sbi\ (C_i)$, as shown below:

$$\underset{j=1}{\overset{|C_i|-1}{\forall}} \left\{ sci\left(C_i\right) \leftarrow \left( abs\left( et(s_{j+1}) - et(s_j) \right) \right) \right\} \quad (4)$$

The whole number of sessions $et(s_j)$, $et(s_{j+1})$ in cluster $C_i$ is represented by the vector $sc_i$, and the exact variance between the session $s_j$ and $s_{j+1}$ completion intervals is transferred to $sci(C_i)$.

3 Begin $pbi(C_i)$ is a vector with the dimensions $|P(C_i)| - 1$ that represents the timestamps of Page Access for a given collection of pages $|P(C_i)|$. Even if multiple sessions are active, the total number of pages is arranged from earliest to latest according to the time of first access. $P(C_i)$ represents the sum of all pages in cluster $C_i$ and is written as a mathematical expression. The shape of the vector $pbi(C_i)$ is depicted below:

$$\underset{j=1}{\overset{|P(C_i)|-1}{\forall}} \left\{ pbi\left(C_i\right) \leftarrow \left( bt(p_{j+1}) - bt(p_j) \right) \right\} \quad (5)$$

By using the cluster $C_i$ and its total number of pages $P(C_i)$, one may correctly reorder the pages $p_j$, $p_{j+1}$ start timings $bt(s_j)$, $bt(s_{j+1})$ on the vector $pbi(C_i)$.

4 For each source cluster $C_i$, the session bandwidth utilization is represented by the vector $bwc(C_i)$, which is of size $|C_i|$. In this case, the "source cluster $C_i$ " displays a total of $C_i$ sessions; the notation "$C_i$ " denotes this total. You can tell how much bandwidth a cluster uses by looking at the total bandwidth consumption for all requests in that session. Each session's bandwidth consumption for the specified source cluster $C_i$ is represented in the following visualization by the vector $wc(C_i)$.

**Algorithm 2:** Bandwidth consumption for Session-level

$$\underset{j=1}{\overset{|C_i|}{\forall}} \left\{ s_j \exists s_j \in C_i \right\}$$

Begin // entire quantity originating from this cluster

$$bwc(C_i) \leftarrow \sum_{k=1}^{|s_j|} \left\{ bw(p_k) \exists p_k \in s_j \right\}$$

// total use of the given bandwidth $bwc(p_k)$

For the resulting session $S_j$, the vector $bwc(C_i)$ is updated with links to all relevant pages $P_k$.

End

### 3.4 Similarity of Cluster Analysis

The first step in bringing this model into action is to sort the resulting clusters so that each cluster's total number of sessions is displayed in descending order. In following this technique, you are at the beginning. The steps that are taken to ensure that clusters are indeed comparable after they have been grouped are as follows:

The steps in determining the degree to which each cluster $ts_i$ is like cluster $r_s$ are depicted below.

Remember the frequency of each trait appearing in the $ts_i, r_s$ clusters.

i. The KS-Test is used to determine the distribution type between the two vectors. The attribute similarity score between the vector and its corresponding clusters $ts_i, r_s$ is 1 if and only if the vector and 0 is clusters share a similar distribution.
ii. Based on the provided average similarity measure of the $ts_i, r_s$ record sets, we may calculate the distribution similarity ratio of the $r_s$ cluster that is connected with $ts_i$.
iii. The cluster group with a similarity ratio of one is found once the similarity score is obtained.
iv. Table 5.2 shows the formula for calculating the distributional similarity of one cluster to another.
v. The vector value presented for relevant features is then taken for each cluster session, and this set $A = \{a_1, a_2, \ldots . a|A|\}$ comprises all such features.

**Algorithm 3:** Assessment of Cluster similarity

$\overset{|T|}{\underset{i=1}{\forall}}\{ts_i \exists ts_i \in T\}$ Begin

// each respective group

$\overset{|T|}{\underset{j=i+1}{\forall}}\{ts_j \exists ts_j \in T\}$ Begin

// for every single one of the other clusters $ts_i$ besides the cluster

$\overset{|A|}{\underset{k=1}{\forall}}\{a_k \exists a_k \in A\}$ Begin

// corresponding to everyone of the features

$v_1 \leftarrow ts_i(a_k)$

$v_2 \leftarrow ts_j(a_k)$

$sim_{ts_i \leftrightarrow ts_j} += ks\_test($

// applying KS-test on vectors $t_j, r_j$ that yields 1 if alike, else yields 0

End

$$dsr_{ts_i \leftrightarrow ts_j} = \frac{sim_{ts_i \leftrightarrow ts_j}}{|A|}$$

// evaluating "the distribution similarity ratio" $dsr_{ts_i \leftrightarrow ts_j}$ amid record sets $ts_i, ts_j$

End

End

### 3.5 Drift Detection Source Cluster Selection Process

Let's assume that the cluster groups set *CG*= $\{cg_1, cg_2, \ldots cg|CG|\}$ represents the expression described in subsection 3.4. Each cluster group, denoted by the notation $\{cg_i \exists cg_i \in CG \wedge 1 \leq i \leq |CG|\}$, represents the cluster set with the transitive property of equality on distribution similarity.

An illustration of each cluster group considered as a source cluster is shown in Table 3. The provided training corpus is used for parallel training, with the instruction being given to both the complete session partitions and the tagged attack vulnerable partitions.

These are the preparation phases: source cluster detection (i.e., choosing the cluster with the highest session count among clusters group with the distribution similarity based on the equal transitive property); lifespan-based session clustering (i.e., grouping sessions that overlap during the active period); similarity detection using the Kolmogorov-Smirnov test; the classifier is trained using the chosen source groups, and the drift is detected; and The AdaBoost algorithm,

**Table 2.** The usage of codes in algorithm 3

| Code | Representation |
|---|---|
| $Ks_{test}$ | Kolmogorov Smirnov test |
| $ts_i ts_j$ | These are grouped items. |
| $Dsr_{ts_i \leftrightarrow ts_j}$ | The proportion of comparable distributions |
| $V_1, V_2$ | These are examples of vectors. |

**Table 3.** The codes used in algorithm 4

| Code | Representation |
|---|---|
| $cg_i$ | *Group cluster* |
| $sc$ | *Source cluster* |
| $csm$ | *count session maximum* |

**Table 4.** The traffic flow statistics were generated

| | Flushing out specifics on the escalating load | Normal workload |
|---|---|---|
| **Session Count** | 47263 | 43484 |
| **Consumption of Bandwidth** | 65230 MB | 42108 MB |
| **Requests count** | 725695 | 726198 |
| **Sources count** | 732 | 518 |

which depends on the throughput of ensemble classification, is one of the paper's innovations.

As stated in subsection 3.3, this research aims to train each classifier using an ensemble of classifiers with optimal properties related to traffic flow. As a result, the suggested method can successfully train any appropriate classifier for use with various classifiers.

## 4 Experimental Study

This section reviews the procedures needed to establish a dataset, develop a strategy, set up an experiment, and compare and contrast the outcomes to determine the strategy's efficacy.

### 4.1 Dataset on DDoS Attack

When assessing attack defence models against DDoS attack record datasets, such as NSL-KDD [29], it is crucial to keep in mind that these datasets contain requests that have been repeatedly iterated, increasing the risk of false positives. This is because these datasets include requests that were created automatically. [24]

This result is necessary, given the evaluation's nature. The three openly accessible databases from DARPA/Lincoln Labs are KDD [31], CAIDA [32], and [33]. The DAPRA datasets span several weeks of network activity and contain data from a simulated Air Force network.

These files contain no sophisticated attacks and have not been altered in any way. According to the study [34], DARPA has distorted the data. Simply sending a user request will grant the user access to the CAIDA dataset. Information about DDoS that happened in 2007 can be found in this CAIDA.

The DDoS attack dataset that CAIDA has only lasted for an hour and includes traffic records from hypothetical situations. Public datasets are unavailable during DDoS attacks because they expose private network information and foreseeable user behaviour. Consequently, we. cannot make these datasets available to the general public. A data leak could result in legal issues and risks to the general public, the organization, and the service suppliers. When applied to simulated data, the outcomes of a detection technique are not accurate. Consequently, it is impossible to compare these results to those of the real-time dataset. The transmission of laboratory data deviates from the anticipated amount of traffic on the upstream routers. According to the study's findings [35], it is impossible to generalize from tiny data samples to more extensive databases.

**Table 5.** Results of the BIFAD experiments

| | |
|---|---|
| Absolute-time intervals Count | 2046(1256:RTF, 790:RTN) |
| measuring intervals in absolute time | 674 (354:RTF, 320:RTN) |
| Intervals of Absolute Time for Physical Training | 1374(749:RTF, 625:RTN) |
| Representations of records that are wholesome | 279 |
| Representations of a record's vulnerability | 326 |
| Salubrious records that have been correctly noted (TN) | 273 |
| Records from Attack Prone have been referred to as salubrious (FN) | 34 |
| Highly vulnerable records have been identified (TP) | 296 |
| Salubrious records have been found in attack-prone databases (FP) | 42 |
| True positive rate(TPR) | 0.923755389 |
| True negative rate(TNR) | 0.904636644 |
| Positive predict value | 0.892151616 |
| Accuracy | 0.898386401 |
| FAR | 0.185749276 |

**Table 6.** The statistics of input given and outcomes achieved from NFBoost

| | |
|---|---|
| Overall Salubrious | 725183 |
| Overall Attack-Prone | 734685 |
| Detection of Positive Results | 264377 |
| Detection of Negative Results | 253528 |
| Training for Positive Results | 520562 |
| Training for Negative Results | 520762 |
| TP | 250968 |
| FP | 31303 |
| TN | 232114 |
| FN | 32518 |
| TPR | 0.892 |
| TNR | 0.8946 |
| predict positive value | 0.8979 |
| accuracy | 0.8997 |
| FAR | 0.1648 |

We evaluated the outcomes of requests produced by DDoSIM, Tor's Hammer, and JMeter. A traffic-generating tool called Tor's Hammer generates lots of traffic while allowing users to publish at their own pace. Consequently, many weak servers are brought down even if they only

**Table 7.** Data on the construction and selection of ECDD source clusters

| | Count of training sessions | Group of Clusters | a single classifier for each cluster of sessions | The number of session clusters | clusters of sources |
|---|---|---|---|---|---|
| **from transaction logs of DDoS attacks** | 34507 | 10 | 11 | 46 | In a range of 721 to 2036 |
| **From Normal Transaction** | 34674 | 7 | 8 | 62 | In a range of 562 to 749 |

have a few users. Based on empirical data, this program demonstrates that standard Apache/IIS servers up to version 6 can abruptly terminate up to 256 threads, whereas versions after this can be compacted by up to 128 threads.

Using the Apache JMeter load-generating tool, which can produce high and varied loads, the performance of the target server or any other server can be assessed in a static or dynamic situation [36].

JMeter, Tor's Hammer, and DDoSIM collected data on regular and attack-prone traffic flows for an hour. To that end, the system was configured to make requests to various sources at regular intervals of 17.6 ms. For instance,Table 4 shows that the attack-prone and typical requests generated 732 malicious and 518 legitimate requests, respectively.

Testing the suggested Ensemble Classifiers with Drift Detection (ECDD)can demonstrate the significance of the claim made in this study and the range of the proposed technique adds to Bio-Inspired Flood Attack Detection (BIFAD)and describes the ensemble of classifiers NFBoost, both of which offer helpful information.

After evaluating each model's performance, the author chooses ECDD and BIFAD; the BIFAD model was chosen due to the effectiveness of its ensemble classifier method. According to research, the BIFAD, because of its distinctive traffic flow characteristics, outperforms the ECDD model in identifying the wide range of DDoS attacks [37]. The results are also contrasted with those from experiments done on the NFBoost model. NFBoostis the most efficient ensemble classifier using the ECDD classifier training method and features of traffic flow traits.

### 4.2 Result Analysis

The experiment is improved in that it validates the ECDD model and evaluates the significance of the outcomes from the BIFAD model and the NFBoost model, respectively, using cross-justification metrics like the true negative rate (TNR), the true positive rate (TPR), and the positive predict value.

Results from high-traffic periods following the implementation of BIFAD are shown in Table 5, where they demonstrate an increase in attack detection precision to 0.89 and a decrease in FAR to 0.12.reports that a small-scale BIFAD model experiment resulted in an attack detection accuracy of 0.96 and a false alarming ratio of 0.045.

As a result, keeping detection accuracy with high traffic volumes and a variety of vehicle types is not particularly dependent on the level of detection accuracy of the BIFAD model. Therefore, the detection accuracy level of the BIFAD model is not especially crucial for preserving detection accuracy.

Additionally, Table 6 displays the outcomes of tests conducted using the NFBoost model. This NFBoost model is trained and assessed using requests rather than sessions in a pre-defined traffic pattern. These figures show that the classifier was trained and evaluated on the real traffic flow at a 70:30 split during the experiment.

According to the findings, the false alarm ratio is 0.12, equal to 12% accuracy, and the detection ratio for the transaction state is 0.89, comparable to 89% accuracy.

The report shows the results of an empirical study on a total of 5150 attack-resistant occurrences and 12950 attackable occurrences.

**Table 8.**The ECDD testing process is provided as inputs

| | |
|---|---|
| Number of times spent training vulnerable positions to attack | 32408 |
| The number of Salubrious training sessions | 31653 |
| Participants in healthy training sessions form groups | 63 |
| groups of possible victims of abuse receiving training | 45 |
| Testing for Negatives | 10613 |
| Testing for Positives | 11548 |
| clusters formed by treatment activities with the objective of evaluation (total negatives) | 31 |
| Attack-prone training sessions are used to form groups (total positives) | 32 |
| Attack-prone groups were incorrectly depicted (False Positives) | 4 |
| Groups with a high potential for attack (True Positives) | 32 |
| Groups that have been misrepresented as negative (False Negatives) | 4 |
| Correctly salubrious groups are represented (True Negatives) | 30 |
| Accuracy | 0.964673 |
| TPR | 0.96 |
| TNR | 0.959342 |
| FAR | 0.09 |
| predict positive value | 0.96 |

According to the research, the NFBoost model had a detection accuracy of 96% and a false alarm rate of 2.8%.

Results from the previous work on the NFBoost (an ensemble classifier model imported from the source) show an unusually high degree of variance compared to the current experimental study.

The practical analysis of NFBoost yields dismal results. Regarding the flow of created traffic, the training method for NFBoost had many examples and a high degree of variation among those examples, both of which added to the algorithm's subpar performance. The ECDD model is also discussed here, and the results obtained are shown in Tables 7-8 and Figure 2.

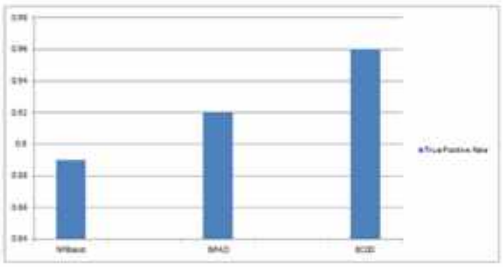
The results of experimental studies used to assess the effectiveness of the ECDD paradigm are shown in Table 8. Compared to the experimental research results, these findings showed that the ECDD model performed better than the BIFAD and NFBoost models in identifying attacks. We discovered that the success rate was 96%, and the erroneous alarm rate was 8%.

This showed the effectiveness of the ECDD model in terms of performance in detecting attacks. The animals have spoken, and BIFAD and NFBoost have an attack detection rate of over 96%.

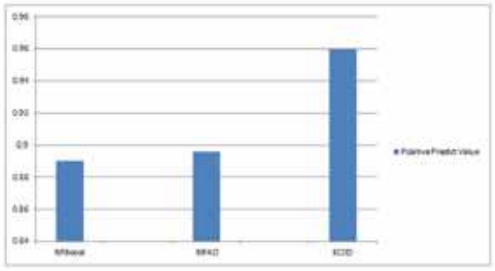
The performance, however, progressively deteriorates when the classifiers are trained on a large volume of traffic and a variety of request types. Using cutting-edge traffic flow features, BIFAD equals the NFBoost ensemble classifier's 96% identification success.

This is made feasible by the distinctive features of the traffic flow at BIFAD. This implies that the two most current models cannot handle large amounts of traffic with various request types.

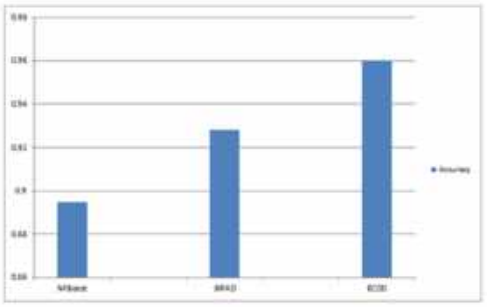
This results in a linear and reasonably quick ECDD processing time for the suggested model.



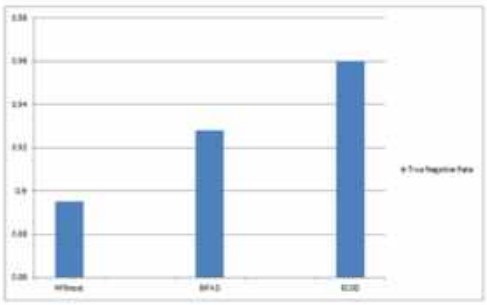
a) True predict value



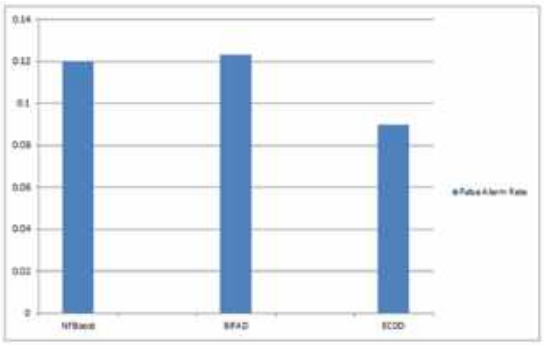
b) Positive predict value



(c) Accuracy



(d) Truenegative rate



e) False alarm rate

**Fig. 2.**The experimental study's presentation statistics are shown in the graphs.

## 5 Conclusion

The computer networks sector has managed the anticipated influx of user requests flooding the internet service. It is now crucial to continuously work on enhancing server performance. Requests that a trustworthy user verifies and those marked by an attacker are the same to the system.

The security of contemporary internet-based services that house servers for various domain platforms, including those used for entertainment and commerce, is seriously threatened by distributed denial-of-service, or DDoS, attacks.

These services offer clients a selection of places ideal for personal and professional activities. It's crucial to practise the defence

mechanism often with heavy traffic and various distribution patterns. The benchmark models used today have so far produced ambiguous findings.

The difference in traffic trends at each node in the distribution network led to inaccurate attack detection. The performance of the current models is compared to a set of benchmark models, which notably feature the BIFAD and NFBoost, to ensure they can handle the massive traffic flow through significant distribution variety qualities. This is done to make sure the current system is current. As a result, experimental research comparisons with other models revealed the scalability and durability of the ECDD model.

Experiments with both cutting-edge and novel models demonstrated that hybrid versions, like BIFAD and ECDD, predicted noticeably better performance to improve the precision of drift and attack detection while reducing the frequency of false alarms.

## References

1. **Patil, N. V., Krishna, C. R., Kumar, K. S. (2022).** SSK-DDoS: distributed stream processing framework based classification system for DDoS attacks. Cluster Computing, Vol. 25, No. 5, pp. 1355–1372. DOI: 10.1007/s10586-022-03538-x.
2. **Jain, M., Kaur, G. (2021).** Distributed anomaly detection using concept drift detection based hybrid ensemble techniques in streamed network data. Cluster Computing, Vol. 24, No. 5, pp. 2099–2114. DOI: 10.1007/s10586-021-03249-9.
3. **Granato, G., Martino, A., Baldini, L., Rizzi, A. (2022).** Intrusion detection in wi-fi networks by modular and optimized ensemble of classifiers: an extended analysis. SN Computer Science, Vol. 3, No. 4, p. 310. DOI: 10.1007/s42979-022-01191-0.
4. **Munivara-Prasad, K., Rama-Mohan-Reddy, A., Venugopal-Rao, K. (2017).** BIFAD: Bio-inspired anomaly based HTTP-flood attack detection. Wireless Personal Communications, Vol. 97, pp. 281–308. DOI: 10.1007/s11277-017-4505-8.
5. **VivinSandar, S., Shenai, S. (2012).** Economic denial of sustainability (EDoS) in cloud services using HTTP and XML based DDoS attacks. International Journal of Computer Applications, Vol. 41, No. 20, pp. 11–16. DOI: 10.5120/5807-8063.
6. **Loukas, G., Öke, G. (2010).** Protection against denial of service attacks: A survey. The Computer Journal, Vol. 53, No. 7, pp. 1020–1037. DOI: 10.1093/comjnl/bxp078.
7. **Linda, O., Vollmer, T., Manic, M. (2009).** Neural network based intrusion detection system for critical infrastructures. 2009 international joint conference on neural networks, IEEE.pp. 1827–1834. DOI: 10.1109/IJCNN.2009.5178592.
8. **Apale, S., Kamble, R., Ghodekar, M., Nemade, H., Waghmode, R. (2014).** Defense mechanism for DDoS attack through machine learning. International Journal of Research in Engineering and Technology, Vol. 3, No. 10, pp. 291–294. DOI: 10.15623/ijret.2014.0310045.
9. **Vijayasarathy, R., Raghavan, S. V., Ravindran, B. (2011).** A system approach to network modeling for DDoS detection using a Naive Bayesian classifier. Third International Conference on Communication Systems and Networks, IEEE, pp. 1–10. DOI: 10.1109/COMSNETS.2011.5716474.
10. **Lu, K., Wu, D., Fan, J., Todorovic, S., Nucci, A. (2007).** Robust and efficient detection of DDoS attacks for large-scale internet. Computer Networks, Vol. 51, No. 18, pp. 5036–5056. DOI: 10.1016/j.comnet.2007.08.008.
11. **Pan, W., Li, W. (2005).** A hybrid neural network approach to the classification of novel attacks for intrusion detection. International Symposium on Parallel and Distributed Processing and Applications, pp. 564–575. DOI: 10.1007/11576235_58.
12. **Pacheco, J., Benitez, V. H., Felix-Herran, L. C., Satam, P. (2020).** Artificial neural networks-based intrusion detection system for internet of things fog nodes. IEEE Access,Vol. 8, pp. 73907–73918. DOI: 10.1109/ACCESS.2020.2988055.

13. **Haddadi, F., Khanchi, S., Shetabi, M., Derhami, V. (2010).** Intrusion detection and attack classification using feed-forward neural network. Proceedings of the 2010 Second International Conference on Computer and Network Technology, IEEE Computer Society, Washington DC, pp. 262–266. DOI: 10.1109/ICCNT.2010.28.
14. **Mendonça, R. V., Teodoro, A. A., Rosa, R. L., Saadi, M., Melgarejo, D. C., Nardelli, P. H., Rodríguez, D. Z. (2021).** Intrusion detection system based on fast hierarchical deep convolutional neural network. IEEE Access, Vol. 9, pp. 61024–61034. DOI: 10.1109/ACCESS.2021.3074664.
15. **Lopez-Martin, M., Sanchez-Esguevillas, A., Arribas, J. I., Carro, B. (2021).** Network intrusion detection based on extended RBF neural network with offline reinforcement learning. IEEE Access, Vol. 9, pp.153153–153170. DOI: 10.1109/ACCESS.2021.3127689.
16. **Albahar, M. A., Binsawad, M., Almalki, J., El-etriby, S., Karali, S. (2020).** Improving intrusion detection system using artificial neural network. International Journal of Advanced Computer Science and Applications, Vol. 11, No. 6. DOI: 10.14569/ijacsa.2020.0110670.
17. **Gupta, B. B., Badve, O. P. (2017).** Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a cloud computing environment. Neural Computing and Applications, Vol. 28, No. 12, pp. 3655–3682. DOI: 10.1007/s00521-016-2317-5.
18. **Huang, Z., Liu, S., Mao, X., Chen, K., Li, J. (2017).** Insight of the protection for data security under selective opening attacks. Information Sciences, Vol. 412, pp. 223–241. DOI: 10.1016/j.ins.2017.05.031.
19. **Gangadharan, K., Kumari, G. R. N., Dhanasekaran, D., Malathi, K. (2020).** Detection and classification of various pest attacks and infection on plants using RBPN with GA based PSO algorithm. Indonesian Journal of Electrical Engineering and Computer Science (IJEECS), Vol. 20, No. 3, pp. 1278–1288. DOI: 10.11591/ijeecs.v20.i3.
20. **Chen, X., Huang, X., Li, J., Ma, J., Lou, W., Wong, D. S. (2015).** New algorithms for secure outsourcing of large-scale systems of linear equations. IEEE Transactions on Information Forensics and Security, Vol. 10, No. 1, p. 38. DOI: 10.1109/TIFS.2014.2363765.
21. **Barford, P., Plonka, D. (2001).** Characteristics of network traffic flow anomalies. Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement, pp. 69–73. DOI: 10.1145/505202.505211.
22. **Kalliola, A., Lee, K., Lee, H., Aura, T. (2015).** Flooding DDoS mitigation and traffic management with software defined networking. Proceedings of 2015 IEEE 4th International Conference on Cloud Networking, IEEE, pp. 248–254. DOI: 10.1109/CloudNet.2015.7335317.
23. **Seufert, S., O'Brien, D. (2007).** Machine learning for automatic defence against distributed denial of service attacks. IEEE International Conference on Communications, pp. 1217–1222. DOI: 10.1109/ICC.2007.206.
24. **Berral, J. L., Poggi, N., Alonso, J., Gavalda, R., Torres, J., Parashar, M. (2008).** Adaptive distributed mechanism against flooding network attacks based on machine learning. Proceedings of the 1st ACM workshop on Workshop on AISec, pp. 43–50. DOI: 10.1145/1456377.1456389.
25. **Huang, G. B., Zhou, H., Ding, X., Zhang, R. (2012).** Extreme learning machine for regression and multiclass classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol. 42, No. 2, pp. 513–529. DOI: 10.1109/TSMCB.2011.2168604.
26. **Srimuang, W., Intarasothonchun, S. (2015).** Classification model of network intrusion using weighted extreme learning machine. 2015 12th International Joint Conference on Computer Science and Software Engineering pp. 190–194. DOI: 10.1109/JCSSE.2015.7219794.
27. **Fossaceca, J. M., Mazzuchi, T. A., Sarkani, S. (2015).** MARK-ELM: application of a novel multiple Kernel learning framework for improving the robustness of network intrusion detection. Expert Systems with Applications,

Vol. 42, No. 8, pp. 4062–4080. DOI: 10.1016/j.eswa.2014. 12.040.

28. **Raj-Kumar, P. A., Selvakumar, S. (2013).** Detection of distributed denial of service attacks using an ensemble of adaptive and hybrid neurofuzzy systems. Computer Communications, Vol. 36, No. 3, pp. 303–319. DOI: 10.1016/j.comcom.2012.09.010.
29. **Revathi, S., Malathi, A. (2013).** A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. International Journal of Engineering Research and Technology. DOI: 10.17762/turcomat.v12i11.6333.
30. **Nazario, J. (2008).** DDoS attack evolution. Network Security, Vol. 2008, No. 7, pp. 7–10. DOI: 10.1016/S1353-4858(08)70086-2.
31. **Shin, S., Gu, G., Reddy, N., Lee, C. P. (2011).** A large-scale empirical study of conficker. IEEE Transactions on Information Forensics and Security, Vol. 7, No. 2, pp. 676–690. DOI: 10.1109/TIFS.2011.2173486.
32. **Sommer, R., Paxson, V. (2010).** Outside the closed world: On using machine learning for network intrusion detection. Security and Privacy (SP), 2010 IEEE Symposium on, pp. 305–316. DOI: 10.1109/SP.2010.25.
33. **Behal, S., Kumar, K. (2017).** Characterization and comparison of DDoS attack tools and traffic generators: A review. International Journal of Network Security, Vol. 19, No. 3, pp. 383–393. DOI: 10.6633/IJNS.201703. 19(3).07.
34. **Badve, O. P., Gupta, B. B. (2016).** Taxonomy of recent DDoS attack prevention, detection, and response schemes in cloud environment. Proceedings of the International Conference on Recent Cognizance in Wireless Communication & Image Processing: ICRCWIP-2014, Springer India, pp. 683–693. DOI: 10.1007/978-81-322-2638-3_76.
35. **Saikam, J., Ch, K. (2024).** An ensemble approach-based intrusion detection system utilizing ISHO-HBA and SE-ResNet152. International Journal of Information Security, Vol. 23, No. 2, pp. 1037–1054. DOI: 10.1007/s10207-023-00777-w.
36. **González, F. J., Aguirre-Anaya, E., Salinas-Rosales, M., Miyaji, A. (2023).** Identification of static and dynamic security controls using machine learning. Computación y Sistemas, Vol. 27, No. 2, pp. 581–592. DOI: 10.13053/CyS-27-2-4429.
37. **Chaparro-Amaro, O. R., Martínez-Felipe, M. D. J., Martínez-Castro, J. A. (2023).** Performance of the classification of critical residues at the interface of BMPs complexes pondered with the ground-state energy feature using random forest classifier. Computación y Sistemas, Vol. 27, No. 1, pp. 257–267. DOI: 10.13053/CyS-27-1-4537.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Dutta Sai Eswari.*

ISSN 2007-9737

# Normal Attractor Intersection Based Multi-objective Optimization Using Particle Swarm Optimization

Josue Dominguez-Guerrero[1,*], Oscar Montiel-Ross[2],
Victor Carrillo[1], Edgar Martinez[1], Oscar Castillo[3]

[1] Universidad Autónoma de Ciudad Juárez,
Mexico

[2] Instituto Politécnico Nacional,
Centro de Investigación y Desarrollo de Tecnología Digital, Tijuana,
Mexico

[3] Instituto Tecnológico de Tijuana,
Mexico

{josue.dominguez, vcarrill, edmartin}@uacj.mx, oross@citedi.mx, ocastillo@tectijuana.mx

**Abstract.** Frequently, multiobjective optimization problems are solved using non-dominated based evolutionary algorithms or gradient-based methods. In the last years, successful proposals that combine the two approaches have been developed. In this work, we propose the Normal Attractor Intersection (NAI) and the NAImopso. The NAI avoids the a priori definition of the search direction and the equality constraints; it uses a set of attractors that cover the entire Pareto Front to generate solutions in the Pareto front. The NAImopso is a multiobjective optimization algorithm based on decomposition; we used it to prove the ability of the NAI to obtain the Pareto front. We compared our proposal against four state-of-the-proposals, and it was evaluated using three well-recognized indicators as performance metrics, the hypervolume indicator, the coverage, and the ε-indicator. The experimental results showed that solutions obtained with the NAImopso were better than the solutions obtained with the other algorithms with it was compared.

**Keywords.** Multi-objetive optimization, MOO classic method, decomposition algorithm, particle swarm optimization.

## 1 Introduction

Optimization problem-solvers aim to find solutions that minimize or maximize at least one objective function (OF). So broadly speaking, optimization problems can be divided according to the number of OF that the solver handles to obtain the optimal values; hence, there are two big groups: In the first one, the problem-solver use only one objective function, thus we are dealing with a single objective optimization problem (SOOP).

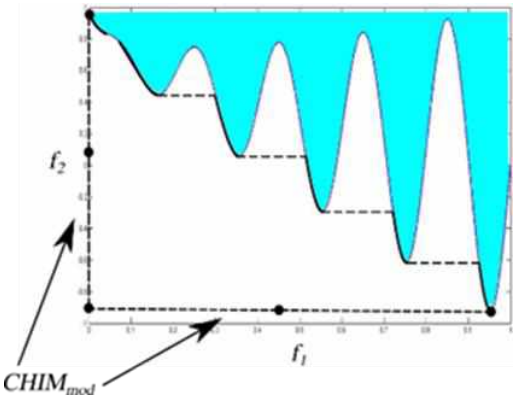
In the second group, the problem-solver compromises the solutions to satisfy in the best possible way more than one OF; these kinds of problems belong to a class known as multiobjective optimization problems (MOOP). MOOPs are harder to solve than SOOPs; consequently, through history, there have been many proposals focused on reducing obstacles in the diverse facets of the problem solver, i.e., the algorithms or methods.

Some ideas of the first proposed methods referred to as classical methods, to distinguish them from metaheuristic-based methods were classified by Miettinen in [1] as follows:

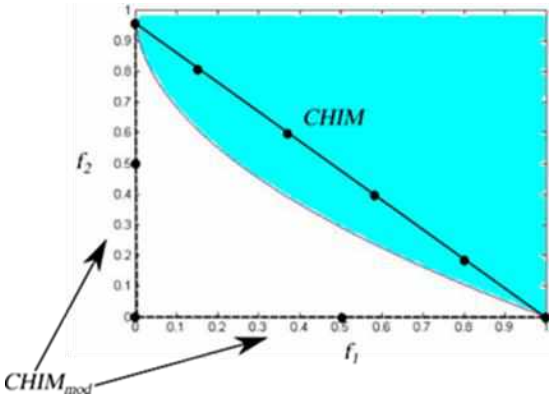
a) No-preference methods.
b) Posteriori methods.
c) A priori methods.
d) Interactive methods.

ISSN 2007-9737






(a)




(b)

**Fig. 1** Graphical representation of the $\mathbf{CHIM_{mod}}$ that cover the entire Pareto front in (a) the zdt1 problem, and (b) zdt3 problem

The no-preference methods use a heuristic to find a single optimal solution without assuming any information concerning the importance of the objectives. In contrast, posteriori methods use preference information about each objective and generate a set of Pareto optimal solutions [2].

The a priori methods usually find a preferred Pareto optimal solution using information concerning the preferences of objectives [3]. Interactive methods work with preference information progressively along the optimization process [4].

Three of the more popular classical methods are the weighted sum approach (WSA), the Benzon method [5], and the normal boundary intersection (NBI) [6, 7].

Regarding metaheuristic methods, Shaffer proposed to solve a MOOP using a genetic algorithm to find a solution in each run of a WSA [8]. A different way to deal with a MOOP is with a multiobjective optimization evolutionary algorithm (MOOEA), which find multiples solutions in the Pareto front in each run.

MOOEA has significantly grown since the Pareto dominance concept was proposed by Goldberg [9] and still being the preferable option to obtain a set of optimal viable solutions. The use of the NBI presents several significant drawbacks; some of them are:

- Transforming a MOOP containing $M$ objectives to a set of single optimization problems with $M$ equality constrains, for problems with a big number of decision variables is more laborious than solving the MOOP.
- In some runs, the NBI returns non-Pareto optimal points. A solution was given in [10], where Logist and Impe, propose a removal criterion for these points using the Lagrange multiplier vectors, then the pay-off matrix and the permutation matrix do not need further comparisons.
- It is necessary to use Robust Parameter Design when solving a MOOP with correlated objective functions [11]. This approach throws unrealistic feasible solutions in the Pareto Front; however, they can be avoided using Principal Component Analysis to obtain uncorrelated objective functions. To deal with correlated objective functions, Dias Lopes et al. [11], proposed the RPD-MNBI that combines Robust Parameter Designs, Principal Component Analysis, and Normal Boundary Intersection preserving the original correlation of the problem and reducing the effects of the noise variables.

A way to deal with the difficulty of the constraint equation in the NBI is combining it with a metaheuristic algorithm and penalize the solution that did not fulfill the constraints like the proposed by Zhang and Li in the MOEA/D [12]. The MOEA/D decompose the objective space of a MOOP in $N$ subproblems with different weight vectors, to solve

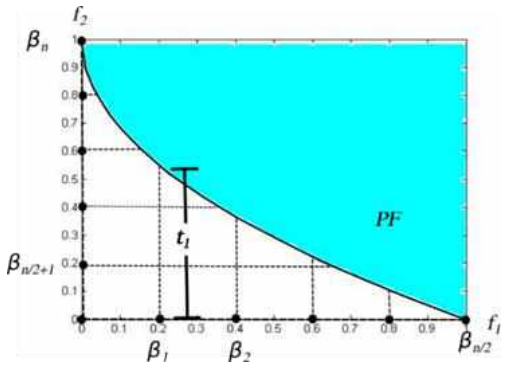



**Fig. 2** Graphical representation of the NAI in the zdt1 problem

these subproblems, the authors reported experiments with three classic methods: The Weighted Sum Approach, the Tchebycheff Approach, and a Boundary Intersection (BI) Approach.

Similarly, in [13], the NBI and the Tchebycheff method are used to guide the search in the neighborhood of the MOEA/D; in this algorithm the classic method and the evolutionary algorithms works together to solve a MOOP; however, the search direction needs to be defined yet.

Another weakness of the MOEA/d is that the recombination operators affect their performance, to solve this in [14] Luo et al. proposed the genetically hybrid differential evolution strategy for recombination that use one mutation strategy focused on the diversity and another focused in convergence.

The problems of the quick convergences and the loss of the diversity affect also the particle swarm optimization algorithms, to avoid this a fusion learning strategy that improved the leadership selection strategy is presented in [15]. The Radial Boundary Intersection based decomposition with an Interior-Point method (RBI-IP) [16] is an algorithm that also uses the NBI.

This algorithm decomposes the objective space in $N$ sub-problems, each of one is solved to find the closets solution to the reference point on the respective radial line, these solutions are found using the interior point method.

Another improvement of the NBI is presented by Cui et al. [17] they use the adaptive weight sum, the adjust uniform axes method and Mahalanobis distance to get a wide and uniform distribution of the Pareto Frontier. In this paper, we propose the Normal Attractor Intersection (NAI) method and the Normal Attractor Intersection multi-objective particle swarm optimization (NAImopso).

The first one is a method inspired by the NBI [6], the improved NBI [18], and the Tchebycheff approach. The NAI avoids the a priori definition of the search direction and the equality constraints; a set of attractors that cover the entire Pareto Front are used, minimizing the normal distance to one of these attractors, then a Pareto point is obtained.

The second one is a multiobjective optimization algorithm based on decomposition; we used it to prove the ability of the NAI to obtain the Pareto front. It guides the search, and the structure of the randomly selected neighbors-particle swarm optimization (RSN-PSO) proposed in [19] adapted to deal with multiobjective optimization is used to generate new solutions.

We compare our proposal against four multiobjective metaheuristics based on the PSO that are in the state-of-the-proposals; they are the pccsAMOPSO [20] and the KGMOPSO [15]. The evaluation was achieved using three well-recognized indicators used as performance metrics; they are the hypervolume indicator, the coverage, and the ε-indicator. The experimental results showed that solutions obtained with the NAImopso were better than the solutions obtained with the other algorithms with it was compared.

The rest of this paper is organized as follows: Section 2 describes the theory of the NBI to understand the inspiration of the NAI. Sections 3 and 4 introduce the proposed NAI and NAImopso respectively. Section 5 reports the experimental results. Section 6 concludes the paper.

## 2 Normal Boundary Intersection

The NBI divides a MOOP with $M$ OFs in subproblems of a SO; each subproblem $\mathrm{NBI}_{\mathbf{w}}$ is solved using different weight vectors $\boldsymbol{w}$ [6].

In the NBI the first step is to find the ideal vector $\mathbf{z}^* = [f_1^*(\mathbf{x}),\ f_2^*(\mathbf{x}),\ \ldots,\ f_M^*(\mathbf{x})]^T$ used in the Convex Hull of Individual Minima (CHIM) described in equation (1):

$$\mathrm{CHIM} = \{\Phi\mathbf{w}\colon \mathbf{w} \in R^M, \sum_{i=1}^{M} w_i = 1, w_i \geq 0\}, \qquad (1)$$

where $\Phi \in \mathrm{R}^{\mathrm{M,M}}$ is a matrix defined in equation (2):

$$\Phi = \begin{matrix} f_1(\mathbf{x_1^*}) - z_1^* & f_1(\mathbf{x_2^*}) - z_1^* & \dots & f_1(\mathbf{x_K^*}) - z_1^* \\ f_2(\mathbf{x_1^*}) - z_2^* & f_2(\mathbf{x_2^*}) - z_2^* & \dots & f_2(\mathbf{x_K^*}) - z_2^* \\ \vdots & \vdots & \ddots & \vdots \\ f_M(\mathbf{x_1^*}) - z_M^* & f_M(\mathbf{x_2^*}) - z_M^* & \dots & f_M(\mathbf{x_K^*}) - z_M^* \end{matrix}, \quad (2)$$

In the CHIM, $n$ indicates the normal direction to it; so, using geometry, the intersection between the normal and the surface of the Pareto front is defined as shown in equation (3) [21]:

$$t\begin{bmatrix} \widehat{n_1} \\ \vdots \\ \widehat{n_M} \end{bmatrix} = \begin{bmatrix} w_1\Phi_{1,1} + \cdots + w_M\Phi_{1,M} \\ w_M\Phi_{M,1} + \cdots + w_M\Phi_{M,M} \end{bmatrix}\begin{bmatrix} f_1(\mathbf{x}) \\ f_M(\mathbf{x}) \end{bmatrix}, \quad (3)$$

where $\boldsymbol{F}(\boldsymbol{x}) = [f_1(x), f_2(x), \dots, f_m(x)]^T$ are the coordinates intersection vector between the normal direction of the CHIM and the Pareto front. Therefore, a MOOP can be converted into a set of subproblems of a SO. The aim is to maximize the distance of $t$ between the normal direction of the CHIM and the Pareto front as follows:

$$\max_{x,t} t, \quad (4)$$

Subject to:

$$\Phi w + t\hat{n} = F(x), \quad (5)$$

$$h_k(x) = 0, k = 1,2, \dots, K, \quad (6)$$

$$g_j(x) \leq 0, j = 1,2, \dots, J. \quad (7)$$

The constraint $\Phi\boldsymbol{w} + t\hat{\boldsymbol{n}} = \boldsymbol{F}(\boldsymbol{x})$ guarantees that the point $\boldsymbol{x}$ mapped by $F(x)$ in the objective space is on the normal direction of the CHIM. Each subproblem is refereed as $\mathrm{NBI}_w$. In the practice instead of $n$, a quasi-normal direction $\hat{n}$ given by (8) is used:

$$\hat{\boldsymbol{n}} = -\Phi\boldsymbol{e}, \quad (8)$$

where $e$ is a column vector of ones.

## 3 Normal Attractor Intersection

The algorithms that transform a MOOP into a single-objective optimization problem are easier to implement and use than the Pareto dominance-based algorithms, but the preference of each objective needs to be defined as a priori in a weight vector.

**Algorithm 1.** NAImopso algorithm

```
1  Initialize the weight vector w ∈ R^{N/M}
2  for each i = 1,2,...,M find the atracctor β^i = {β_1^i, β_2^i, ..., β_{N/M}^i}
3  Initialize the swarm p^1, ..., p^N and assign an attractor to each one
4  for each particle p^i in the swarm initialize a sub-warm with τ particles
5  for each particle p^i find the τ closest attractors and assign one of these attractors to each particle in the sub-swarm of p^i
6  while Stop criteria is not fulfilled do
7      for p ← 1, N
8          for i ← 1, τ
9              if stagnation(i) ≥ renewalGap then
10                 Select randomly n_s particles from the subswarm
11                 Update RSNbest with the particle with best fitness of the previously selected
12                 stagnation(i) = 0
13             end if
14             update velocity and position using the equations (7) and (8)
15             evaluate each objective function for the new position.
16             update the particle fitness using the NAI
17             if fitness of particle i is better than fitness of Pbest then
18                 update Pbest
19             else
20                 stagnation(i) = stagnation(i) + 1
21             end if
22             if fitness of particle i is better than fitness of Sbest then
23                 update Sbest
24             end if
25         end for
26     end for
27 end while
```

The NBI has advantages over other classical methods that obtain solutions with a better distribution on the Pareto front; in contrast, it converts a MOOP in a set of SO optimization problems with equality constraints that could be hardest to solve than the MOOP. In gradient-based methods, the Karush-Kuhn-Tucker condition is used to deal with equality constraints. In evolutionary algorithms, it is common to penalize

**Table 1.** Comparative of the NAImopso, pccsAMOPSO,and KGMOPSO algorithms using the IGD in the ZDT1, ZDT2, ZDT3, ZDT4 and DTLZ2 problems

| Functions | | NAImopso | pccsAMOPSO | KGMOPSO |
|---|---|---|---|---|
| ZDT1 | Mean | **2.20E-3** | 4.01E-3 | 4.04E−3 |
| | Std. | 1.15E-3 | 6.28E-5 | 8.23E−5 |
| ZDT2 | Mean | **5.45E-5** | 4.09E-3 | 3.98E−3 |
| | Std. | 1.66E-1 | 4.81E-5 | 4.99E−5 |
| ZDT3 | Mean | 1.30E-2 | **3.32E-3** | 5.51E−3 |
| | Std. | 5.53E-3 | 9.95E-5 | 9.67E−5 |
| ZDT4 | Mean | 2.3E-1 | 7.97E-3 | **4.21E−3** |
| | Std. | 1.62E-1 | 1.47E-3 | 7.13E−5 |
| ZDT6 | Mean | 3.33E-2 | 3.4E-3 | **3.39E−3** |
| | Std. | 2.09E-2 | 2.28E-4 | 1.46E−4 |
| DTLZ2 | Mean | **1.80E-3** | 6.14E-2 | 7.48E−2 |
| | Std. | 8.71E-6 | 1.89E-3 | 2.50E−3 |

**Table 2.** Comparative of the NAImopso and MOEA/d algorithms using the $\boldsymbol{S}$, $\mathbf{0}$ y $\boldsymbol{I}_{\epsilon+}$ metrics in the the ZDT1, ZDT2, ZDT3 and DTLZ2 problems

| A= NAImopso y B=MOEA/d | | | | |
|---|---|---|---|---|
| **Metric** | **ZDT1** | **ZDT2** | **ZDT3** | **DTLZ2** |
| $S(A)$ | 3.7865 | **3.7802** | **5.3166** | 2.8787 |
| $S(B)$ | **3.9638** | 1.9843 | 3.1667 | **3.9998** |
| $C(A,B)$ | **0.4455** | **0.3069** | **0.5940** | **0.6079** |
| $C(B,A)$ | 0.0099 | 0 | 0.3964 | 0 |
| $I_{\epsilon+}(A,B)$ | 0.1358 | **0.0271** | 0.0559 | 1 |
| $I_{\epsilon+}(B,A)$ | **0.0079** | 0.1095 | **0.0144** | **0.0949** |

the individuals that do not satisfy the constrains; an example is the penalty-based boundary intersection (PBI) approach [12].

In the NBI, the search direction needs to be defined. If the objective space has a convex region and a non-convex region, the direction needs to change. In [18] use the $\text{CHIM}_{\text{MOD}}$, a dashed line over the axis $f_1(x)$, the problem of maximize $t$ in the original NBI is changed to minimize $t$,and the search direction is not dependent of the objective space form, in [18] only are reported results for bi-objective problems.

In our proposal, the Normal Attractor Intersection (NAI) method, the normal vector direction to CHIM, does not have to be defined a priori, which is an advantage over the NBI. The above advantage is due to the NAI uses a set of attractors $\{\beta^1, \beta^2, \dots, \beta^N\}$, where $\beta^n \in R^M$, instead of using the CHIM. The attractors enclose the entire Pareto front, as shown in the **¡Error! No se encuentra el origen de la referencia.**, so the set of attractors is a set of points distributed in M lines parallel to the objective axis.

Equation (9) allows to obtain each element $\beta_{\mathrm{m}}^{\mathrm{n}}$ of an attractor point $\beta^n = [\beta_1^n, \beta_2^n, \dots, \beta_M^n]^T$ parallel to the i objective axis. Where $w_n \in [0,1]$ is the weight of the attractor point n. The equation (9) only applies for $m = i$. For $m \neq i$, $\beta_m^n = z_m^*$, this is to ensure that the attractor point n is parallel to the objective axis i, and it covers all the Pareto front:

$$\beta_i^n = w_n(max(F(x)_k^*) - min(F(x)_k^*)); \quad \forall k = 1,2,\dots,M;\ k \neq i. \tag{9}$$

Using the set of attractors, the NAI convert a MOOP in $N$ subproblems of one objective were to obtain a point in the Pareto front the value of the normal distance $t_n$ between the objective space and the attractor $\beta^n$ is minimized like is shown in the Fig. 2, so using this approach a $n$ subproblem of the MOOP is solving as follows:

Minimize:

$$t_n. \tag{10}$$

Subject to:

$$\beta^n + t\hat{n} = F(x), \tag{11}$$

$$g_j(x) \geq 0, j = 1,2,\dots,J, \tag{12}$$

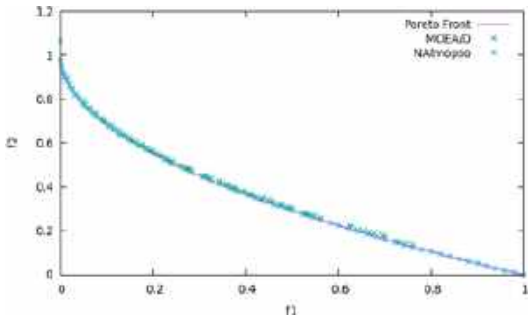
$$h_k(x) = 0, k = 1,2,\dots,K, \tag{13}$$

where $\hat{n}$ is the normal vector obtained by the equation (14) and $e$ is a vector of ones:

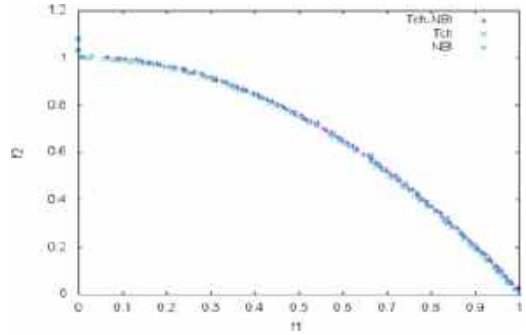
$$\hat{n} = -\hat{n} = -\Phi e. \tag{14}$$

The previous approach has the same problem of the $NBI$ and the improved $NBI$; the equality constraints still need to be satisfied.

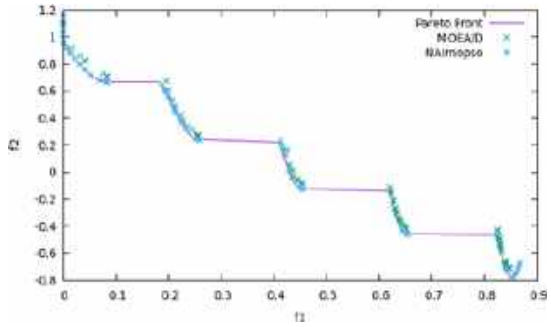
For these reasons, we propose to use a method like Tchebycheff decomposition, but instead of minimizing the maximal distance between a point and the ideal vector multiplied by a weight, we minimize the maximal distance between a point and an attractor.




(a)




(b)




(c)

**Fig. 3** We indicated the True Pareto front using a `-', the Pareto approximate front obtained with the MOEA/D with an 'x', and the Pareto approximate front obtained with the NAImopso with an '*'. (a) Pareto front of the ZDT1. (b) Pareto front of the ZDT2. (c) Pareto front of the ZDT3

To ensure that the obtained Pareto optimal point is parallel to the attractor, the angle $\alpha$ penalize the solution, while $\alpha$ is closer to zero, the point is closer to the normal, so $\alpha t_n$is an image of $t_n$ in the previous approach, this new method is called Normal Attractor Intersection NAI, so a MOOP is decomposed in $n$ subproblems, each of one is solved as follow:

Minimize:

$$t_n = \alpha \max_m^M \{|f_m(x) - \beta_m^n|\} \tag{15}$$

Subject to:

$$g_j(x) \geq 0, j = 1,2,\dots,J, \tag{16}$$

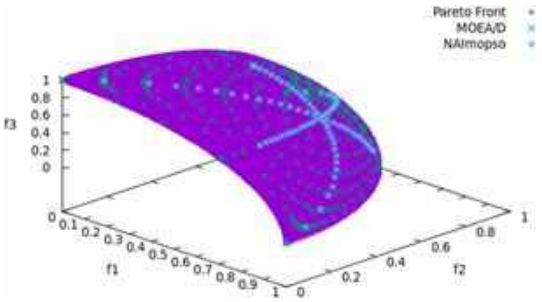
$$h_k(x) = 0, k = 1,2,\dots,K, \tag{17}$$

where $\alpha = \operatorname{atan}\left(\frac{f_i(x)-\beta_i^n}{f_k(x)-\beta_k^n}\right) > 0$ is a variable to keep the solution normal to the $i$ axis, and $f_k(x) - \beta_k^n$ is the distance of the solution in any $k \neq i$ axis to the attractor point.

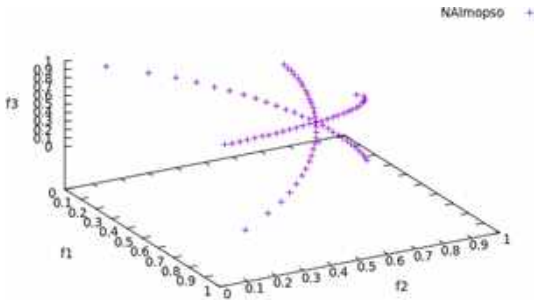
## 4 Normal Attractor Intersection Based Multi-objective Optimization Using Particle Swarm Optimization NAImopso

The NAImopso is a PSO based algorithm that decomposes the objective space of a MOOP in $N$ subproblems; each one is solved using the NAIB with different attractors. The fitness of each particle is obtained evaluating one subproblem, so the size of the swarm is $N$ too; each particle $p$ has an attractor $\beta^p$ and a neighborhood $B^p$, where $B^p$ is a set with the $\tau$ closets attractors to $\beta^p$, for each particle $p$ a new subswarm is generated with $\tau$ particles that are evaluated with one of the attractors in $B^p$.

To preserve the diversity and ensure to obtain a good approximation to the Pareto front, the particles in NAImopso learn from Sbest which is the particle with the best fitness in the subswarm and from RSNbest that is the best particle in a randomly selected neighborhood of size $n_s$ from the subswarm. The use of the RSNbest was proposed in [19] were to avoid the stagnation used the renewelGap parameter, which is the maximal number of iterations without update the Sbest.




(a)




(b)

**Fig. 4** Non-dominated and Pareto front candidates obtained with the MOEA/D and the NAImopso for DTLZ2 function. (a) Pareto front of the DTLZ2. (b) Pareto front of the DTLZ2 obtained with the NAImops

The position and velocity of each particle $i$ in the subswarm is updated using the equations (18) and (19) where $c_1$, $c_2$ and $c_3$ are the coefficients, $Sbest_i^d(t)$ is the decision variable $d$ of the particle in the subswarm with the best fitness, and $RSNbest_i^d(t)$ is the value of the best particle in the random selected neighborhood:

$$v_i^d(t+1) = X\Big(v_i^d(t) + c_1 r_1\left(\text{Pbest}_i^d(t) - x_i^d(t)\right) + c_2 r_2\left(\text{RSNbest}_i^d(t) - x_i^d(t)\right) + c_3 r_3\left(\text{Sbest}_i^d(t) - x_i^d(t)\right)\Big), \quad (18)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \quad (19)$$

To calculate the fitness of the new position $x_i(t+1)$ of each particle $i$ first each objective function is evaluated with this position, $F_1\big(x_i(t+1)\big)$, $F_2\big(x_i(t+1)\big)$, … , $F_M\big(x_i(t+1)\big)$, then a scalar fitness is calculated using the NAIB with the attractor $\beta^i \in B^p$ previously assigned.

If the fitness of the new position is minimal, assuming that it is a minimization problem, the Pbest fitness and the Sbest fitness are updated. The NAImopso pseudocode is shown in Algorithm 1; the output is the Pbest of each particle, i.e., $N$ points distributed in the Pareto front.

## 5 Experiments

To test the performance of the proposed NAImopso the two objective benchmark function ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6 [22] and the three-objective function DTLZ2 [23] were used. We compared the results against other state-of-the-art multiobjective metaheuristics based on the PSO; they are the pccsAMOPSO [20] and the KGMOPSO [15]. In the tests performed with the NAImopso, the number of particles used was $N = 100$. This number is equal to the numbers of attractors; for each one, we have a point in the Pareto front.

The number of particles in the subswarm of each particle was $\tau = 40$; the size of the subswarm was selected according to the most common size reported in the literature; we obtained good results in performance and computational time with this size.

A value of the size of the neighborhood that can obtain a good balance between exploration and exploitation, according to [19] is $n_s = 5$. The parameters of the PSO were set as $X = 0.7298$ and $c_1 = c_2 = c_3 = 2.05$.

The experiments were running in a computer with the processor Intel(R) Core (TM) i7-8650U CPU @ 1.90GHz and 16 Gb of RAM. The algorithm was implemented in C++, the mean execution time for each run was 165.21 seconds, the faster time was 44.0 seconds and the slower was 239.0 seconds.

The inverted generational distance (IGD) is used to compare the performance of the NAImopso, the IGD is a measure between the solution $S$ and a set of targeted points on the Pareto Front $R$ [24]. The IGD is calculated as follows:

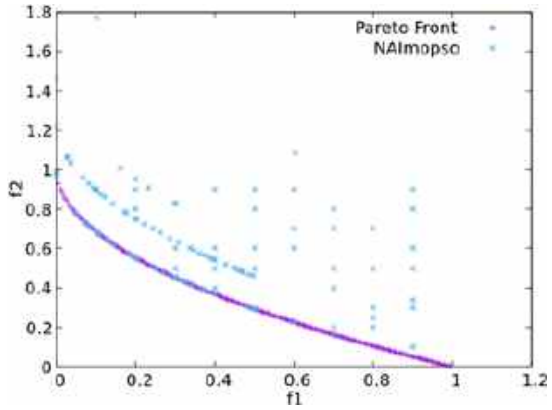



**Fig. 5** Final solutions of the NAImopso for the ZDT4 problem

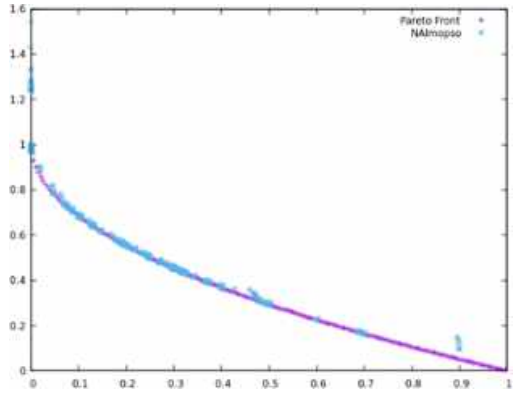



**Fig. 6** Solutions of the NAImopso for the ZDT4 problem after eliminated de dominated solutions

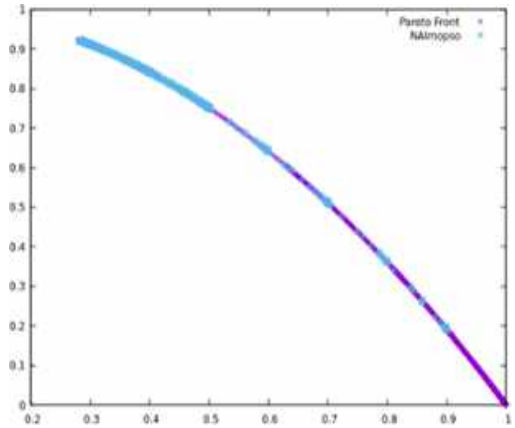



**Fig. 7** Solutions of the NAImopso for the ZDT6 problem after eliminated de dominated solutions

$$IGD(s,R) = \frac{1}{|R|}\sum_{i=1}^{|R|} \min_{j=1}^{\mu} \left(d(r_i, s_j)\right), \qquad (20)$$

where $d(r_i, s_j)$ is the euclidean distance between the solution $S \in S_1, S_2, \ldots, S_\mu$ and the targeted point $R \in R_1, R_2, \ldots, R_{|R|}$.

We executed the NAImopso 30 times to obtain the mean and the standard deviation of the IGD. The results and comparisons are shown in the Table 1, we can observe that the NAImopso obtained better results than other algorithms when optimizing the ZDT1, ZDT2, and DTLZ2 problems.

The NAImopso does not use any non-dominated method, because of that for multimodal problems as the ZDT4 and non-uniform (ZDT6) all the solution is not in the Pareto Front, but with a non-dominated method in the last population we can remove the dominated solutions. There are different metrics (indicators) to evaluate the output quality or the different evaluated algorithms [25, 22]. In this work, we chose to perform the evaluation using the next three metrics: the hypervolume indicator $S(X')$ [26], the additive epsilon indicator $I_{\epsilon+}(X', X'')$ [26], and the coverage $C(X', X'')$ metric [27].

The hypervolume indicator is proportional to the distribution of the solutions $X'$ in the Pareto front; hence this metric captures the proximity of the solution set to the true Pareto front as well as its distribution in the objective space, better solution sets are those with the larger values. The additive epsilon indicator measures the smallest distance by which a Pareto approximate front ($X'$), obtained by one algorithm, must be shifted in the objective space to dominate another Pareto approximate front ($X''$) obtained by another algorithm.

It provides a relative measurement that expresses the minimum necessary value of $\epsilon$ that should be added to the solutions $x'_i \in X'$, and then they dominate all solutions in $X''$ [28]; the smaller values represent the best solutions.

The coverage $C(X', X'')$ indicator describes the percentage of solutions in the set $X'$ that dominates the solutions in the set $X''$; in this case, it is desirable to get big values from this indicator [27]. **Table 2** shows the results of these evaluations. In the table, we can observe that the hypervolume value of the solutions obtained with the NAImopso $S(A)$ is better for the ZDT2 and ZDT3 problems.

In the four problems, the solutions of the NAImopso cover a larger percentage of the solutions of the MOEA/d $C(A,B)$; conversely, the MOEA/d cannot cover the range of solutions obtained with the NAImopso $C(B,A)$. Additionally, according to the additive epsilon indicator, the NAImopso is better in the ZDT2 problem. In the Fig. 3 is shows the Pareto front of the ZDT1, ZDT2, and ZDT3 problems, the non-dominated points obtained with the MOEA/D and the Pareto candidates obtained with the NAImopso.

In Fig. 3 (a) and Fig. 3 (b) show the results obtained for the ZDT1 and ZDT2 functions, we can observe that the solutions of the NAImopso have a better distribution on the Pareto front than MOEA/d, the above can be corroborated because the value of the $S$ metric is higher as can be seen in **Table 2** which also shows that the coverage of the NAImopso over the MOEA/d is therefore higher. In Fig. 3 (c) we can observe that the solution of the NAImopso is closer to the Pareto front but to there are a set of points that no are non-dominated solutions.

In Fig. 4 (a) the Pareto front of the DTLZ2 function, the non-dominated points obtained with the MOEA/D, and the Pareto candidates obtained with the NAImopso are shown, the results of the NAImopso are on the Pareto front but as can see in the Fig. 4 (b) are only the solutions normal to the axis and not the entire Pareto front.

For the ZDT4 problem due to it is multimodal, in the final solution there are non-dominated and dominates solutions, as shown in the Fig. 5, in the solution of the NAImopso we use the algorithm proposed by Mishra and Harit [28] to find the non-dominated solutions, the final population is the show in the Fig. 6. For the ZDT6 problem, we also used the Mishra and Harit algorithm to keep only the non-dominated solutions, as shown in Fig. 7.

## 6 Conclusion

In this paper, we proposed the NAI and the NAImopso methods; they provide a synergetic combination of a classical method and a bio-inspired algorithm that performs better than the MOEA/d. The combination of methods allows obtaining good results without using the non-dominated search that uses most of the Evolutionary algorithms. The solutions are Pareto candidates, and we do not use any strategy to ensure that the solutions are Pareto points; however, the decision-maker could apply a non-dominated method or evaluate only the selected solutions.

For problems with three objectives, a good representation of Pareto front is obtained which in real multi-objective optimization problems could facilitate the work for the decision-maker, although the solutions are not distributed in the entire Pareto front. We compared the NAImopso with other PSO multi-objective algorithms, such as the pccAMOPSO, cdAMOPSO, clusterMOPSO, and the pdMOPSO. The IGD metric was used to evaluate the different algorithms. The NAImopso obtained better solutions when optimizing the ZDT1, ZDT3, and DTLZ2 problems. In the case of the ZDT2, the results of NAImopso were as good as those obtained with the other algorithms.

Independently of the NAImopso, the NAI can be used with other multi-objective algorithms to guide the search. Moreover, it can be used in preference algorithms and to optimize problems with many-objectives. Also, to improve the capacity to find better non-dominated solutions we can prove to add dynamic adjustment of the parameters with fuzzy logic like the proposed in [29] or use NAImopso with a Bacterial foraging optimization (BFO) as is described in [30].

## Acknowledgments

We thank to the Universidad Autónoma de Ciudad Juárez, the Instituto Politécnico Nacional and the Consejo Nacional de Ciencia y Tecnología for supporting our research activities.

## References

1. **Miettinen, K. (1998).** Nonlinear multiobjective optimization. International Series in Operations Research and Management Science, Springer US, Vol. 12, pp. 257–298. DOI: 10.1007/978-1-4615-5563-6.
2. **Stadler, W. (1979).** A survey of multicriteria optimization or the vector maximum problem, part I: 1776–1960. Journal of Optimization

Theory and Applications, Vol. 29, No. 1, pp. 1–52. DOI: 10.1007/bf00932634.

3. **Rivera, G., Coello-Coello, C. A., Cruz-Reyes, L., Fernandez, E. R., Gomez-Santillan, C., Rangel-Valdez, N. (2022).** Preference incorporation into many-objective optimization: an ant colony algorithm based on interval outranking. Swarm and Evolutionary Computation, Vol. 69, pp. 101024. DOI: 10.10 16/j.swevo.2021.101024.
4. **Rivera, G., Cruz-Reyes, L., Fernandez, E., Gomez-Santillan, C., Rangel-Valdez, N. (2023).** An interactive ACO enriched with an eclectic multi-criteria ordinal classifier to address many-objective optimisation problems. Expert Systems with Applications, Vol. 232, pp. 120813. DOI: 10.1016/j.eswa. 2023.120813.
5. **Benson, H. P. (1992).** A finite, nonadjacent extreme-point search algorithm for optimization over the efficient set. Journal of Optimization Theory and Applications, Vol. 73, No. 1, pp. 47–64. DOI: 10.1007/bf00940077.
6. **Das, I., Dennis, J. (1996).** Normal-boundary intersection: An alternate method for generating pareto optimal points in multicriteria optimization problems. Contractor, no. 96, pp. 38.
7. **Ganesan, T., Vasant, P., Elamvazuthi, I. (2013).** Normal-boundary intersection based parametric multi-objective optimization of green sand mould system. Journal of Manufacturing Systems, Vol. 32, No. 1, pp. 197–205. DOI: 10.1016/j.jmsy.2012.10.004.
8. **Schaffer, J. D. (1985).** Some experiments in machine learning using vector evaluated genetic algorithms, PhD Thesis.
9. **Goldberg, D. E. (1989).** Genetic algorithms in search, optimization and machine learning, 1st ed. Addison-Wesley Longman Publishing Co., Inc.
10. **Logist, F., van-Impe, J. (2011).** Novel insights for multi-objective optimisation in engineering using normal boundary intersection and (enhanced) normalised normal constraint. Structural and Multidisciplinary Optimization, Vol. 45, No. 3, pp. 417–431. DOI: 10.1007/s00 158-011-0698-8.
11. **Dias-Lopes, L. G., Brito, T. G., Paiva, A. P., Peruchi, R. S., Balestrassi, P. P. (2016).** Robust parameter optimization based on multivariate normal boundary intersection. Computers and Industrial Engineering, Vol. 93, pp. 55–66. DOI: 10.1016/j.cie.2015.12.023.
12. **Zhang, Q., Li, H. (2007).** MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation, Vol. 11, No. 6, pp. 712–731. DOI: 10.1109/tevc.2007.892759.
13. **Rubio-Largo, Á., Zhang, Q., Vega-Rodríguez, M. A. (2014).** A multiobjective evolutionary algorithm based on decomposition with normal boundary intersection for traffic grooming in optical networks. Information Sciences, Vol. 289, pp. 91–116. DOI: 10.1016/j.ins.2014.08.004.
14. **Luo, N., Lin, W., Jin, G., Jiang, C., Chen, J. (2021).** Decomposition-based multiobjective evolutionary algorithm with genetically hybrid differential evolution strategy. IEEE Access, Vol. 9, pp. 2428–2442. DOI: 10.1109/access. 2020.3047699.
15. **Li, W., Meng, X., Huang, Y., Mahmoodi, S. (2021).** Knowledge-guided multiobjective particle swarm optimization with fusion learning strategies. Complex and Intelligent Systems, Vol. 7, No. 3, pp. 1223–1239. DOI: 10.1007/s40747-020-00263-z.
16. **Datta, S., Ghosh, A., Sanyal, K., Das, S. (2017).** A radial boundary intersection aided interior point method for multi-objective optimization. Information Sciences, Vol. 377, pp. 1–16. DOI: 10.1016/j.ins.2016.09.062.
17. **Cui, J., Pan, J., Wang, S., Okoye, M. O., Yang, J., Li, Y., Wang, H. (2022).** Improved normal-boundary intersection algorithm: a method for energy optimization strategy in smart buildings. Building and Environment, Vol. 212, pp. 108846. DOI: 10.1016/j.buildenv. 2022.108846.
18. **Siddiqui, S., Azarm, S., Gabriel, S. A. (2012).** On improving normal boundary intersection method for generation of pareto frontier. Structural and Multidisciplinary Optimization, Vol. 46, No. 6, pp. 839–852. DOI: 10.1007/s00 158-012-0797-1.

19. **Sun, W., Lin, A., Yu, H., Liang, Q., Wu, G. (2017).** All-dimension neighborhood based particle swarm optimization with randomly selected neighbors. Information Sciences, Vol. 405, pp. 141–156. DOI: 10.1016/j.ins.2017.04.007.
20. **Hu, W., Yen, G. G. (2015).** Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system. IEEE Transactions on Evolutionary Computation, Vol. 19, No. 1, pp. 1–18. DOI: 10.1109/tevc.2013.2296151.
21. **Aghaei, J., Akbari, M. A., Roosta, A., Baharvandi, A. (2013).** Multiobjective generation expansion planning considering power system adequacy. Electric Power Systems Research, Vol. 102, pp. 8–19. DOI: 10.1016/j.epsr.2013.04.001.
22. **Zitzler, E., Deb, K., Thiele, L. (2000).** Comparison of multiobjective evolutionary algorithms: empirical results. Evolutionary Computation, Vol. 8, No. 2, pp. 173–195. DOI: 10.1162/106365600568202.
23. **Deb, K., Thiele, L., Laumanns, M., Zitzler, E. (2002).** Scalable multi-objective optimization test problems. Proceedings of the Congress on Evolutionary Computation, Vol. 1, pp. 825–830. DOI: 10.1109/cec.2002.1007032.
24. **Narukawa, K., Setoguchi, Y., Tanigaki, Y., Olhofer, M., Sendhoff, B., Ishibuchi, H. (2015).** Preference representation using gaussian functions on a hyperplane in evolutionary multi-objective optimization. Soft Computing, Vol. 20, No. 7, pp. 2733–2757. DOI: 10.1007/s00500-015-1674-9.
25. **Okabe, T., Jin, Y., Sendhoff, B. (2003).** A critical survey of performance indices for multi-objective optimisation. The Congress on Evolutionary Computation, Vol. 2, pp. 878–885. DOI: 10.1109/cec.2003.1299759.
26. **Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., da-Fonseca, V. (2003).** Performance assessment of multiobjective optimizers: an analysis and review. IEEE Transactions on Evolutionary Computation, Vol. 7, No. 2, pp. 117–132. DOI: 10.1109/tevc.2003.810758.
27. **Zitzler, E., Thiele, L. (1999).** Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation, Vol. 3, No. 4, pp. 257–271. DOI: 10.1109/4235.797969.
28. **Mishra, K. K., Harit, S. (2010).** A fast algorithm for finding the non dominated set in multiobjective optimization. International Journal of Computer Applications, Vol. 1, No. 25, pp. 46–54. DOI: 10.5120/460-764.
29. **Kawano, Y., Valdez, F., Castillo, O. (2022).** Fuzzy combination of moth-flame optimization and lightning search algorithm with fuzzy dynamic parameter adjustment. Computación y Sistemas, Vol. 26, No. 2, pp. 743–757. DOI: 10.13053/cys-26-2-4269.
30. **Mathur, R. P., Sharma, M. (2023).** A multi-objective task scheduling scheme GMOPSO-BFO in mobile cloud computing. Computación y Sistemas, Vol. 27, No. 2, pp. 477–488. DOI: 10.13053/cys-27-2-3953.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Josue Dominguez-Guerrero.*

ISSN 2007-9737

# Prescriptive Analytics on Cloud Based Systems Using Deep Learning Techniques

Srinivas Naveen Dolu-Surabhi[1,*], Manikandan Rajagopal[2], M. Hema Kumar[3], S Srividhya[4]

[1] General Motors, Michigan, United States of America

[2] School of Business and Management, Lean Operations and Systems, Bangalore, India

[3] College of Technology, Department of ECE, Sona, Salem, Unites States of America

[4] KPR College of Arts Science and Research, Department of Computer Science, Coimbatore, India

{srinivas.csii, mani4gift, vidhyasai14}@gmail.com, hemakumar.ece@sonatech.ac,

**Abstract.** Today's cloud-based systems, often called Cloud Computing, require robust Prescriptive Analytics to optimize the efficacy of Decision-Making processes. This study addresses the problems facing Prescriptive Analytics by implementing a fresh approach that utilizes Deep Auto Encoder Optimisation. The lack of approaches integrating Cloud Computing - Prescriptive Analytics systems with Deep Autoencoder optimization creates a field of study. Such methods make this an issue. Our strategy leverages Deep Autoencoders to uncover complex patterns and correlations from various datasets, improving the precision and effectiveness of Cloud Computing - Prescriptive Analytics systems. The outcome indicates that the DM procedures were greatly improved because of the substantial quantity of research and validation that has to be performed, which proves that the technique is effective. In addition to resolving the problems with Cloud Computing- Prescriptive Analytics, Decision-Making process optimization paves the way for potential improvements in Artificial Intelligence-powered decision support systems.

**Keywords.** Cloud-based systems or cloud computing, prescriptive analytics, decision-making, deep autoencoder optimization, artificial intelligence

## 1 Introduction

The development of Cloud Computing (CC) has resulted in a modern phase of data-driven Decision-Making (DM) [1]. When implemented in cloud settings, standard analytics methodologies encounter challenges due to the complex and dynamic nature of CC [2].

Optimizing the outcomes of decisions is an evident objective of effective Deep Autoencoder (PA) in these circumstances [3].

The challenges of handling enormous, heterogeneous data complexities within cloud environments might exceed the current approaches' capabilities [4].

This is particularly valid in the case of decision support. Maintaining the dynamic nature of CC is challenging due to the limitations of conventional analytics methodologies [5].

There are several methods available to address those limitations. New and innovative methods are required to overcome those risks and thereby enhance the effectiveness and precision of PA [6].

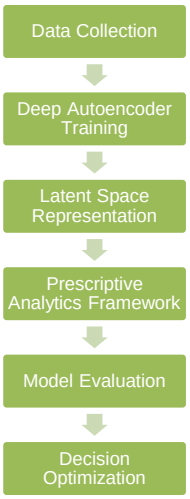



**Fig. 1.** Proposed method

The current situation is the absence of methodologies that can rapidly integrate complex optimization methods into cloud-hosted PA systems [7, 8, 9]. This study presents a technique to identify complex correlations and patterns in data stored in clouds to meet the demand. Improving DM processes and addressing shortcomings in PA that are presently offered on cloud services.

Our proposal offers an innovative approach to decision support systems by bringing DAE optimization methods into the CBA (Cloud-Based Analytics) realm. This work contributes to establishing the foundations for future improvements in Artificial intelligence-driven cloud-based PA.

This research has benefits over its initial application. The goal of this study is to determine new methods that can be applied to maximize outcomes of decisions in the dynamic and constantly evolving field of CC by using.

## 2 Related Works

Previous studies examined several aspects of cloud-based PA, emphasizing this ever-evolving subject's potential advantages and risks [10]. Conventional analytics techniques have found that conventional methodologies cannot handle the dynamic CC complexity [11]. The results of these studies [12] have highlighted the importance of creating novel strategies to improve cloud-based DM procedures.

Numerous studies were carried out on the complexity of big datasets in cloud environments, and the difficulties encountered in extracting pertinent information were reported [13]. New methods are established as an outcome of a study into how standard analytics cannot handle the specifics of information stored in the cloud [14, 15].

To obtain the greatest possible optimization outcomes, scholars have looked into several cutting-edge methodologies, one of which is DAE optimization. These research results show how much Auto Encoders (AEs) can be used to uncover relationships and patterns in various datasets. Despite this, there is still no research on integrating these optimization strategies into cloud-hosted PA systems [16, 17, 18, 19, 20].

The literature highlights the importance of broadening current perspectives to find new applications for cloud-based decision support systems. This compilation of significant publications provides the foundation for the current investigation by providing a structure for comprehending the state of the art and limitations in cloud-based PA.

## 3 Proposed Method

To get around the drawbacks of standard PA in CC, the suggested methodology offers an original approach incorporating DAE optimization. As shown in Figure 1, the principal objective is to improve DM by using DAEs to obtain complex correlations and patterns from large datasets saved in the cloud.

Preprocessing is done on several datasets to prepare them for integration into the DAE framework. To provide a more complete image, the deep autoencoder is specifically intended to obtain and encode the hidden properties of the input. The AE utilizes an iterative approach to its settings to accurately repeat the input information and identify significant patterns. Once the DAE has finished training, the encoded patterns by the PA architecture will be. Information that up is put into

**Table 1**. Description of the real-world dataset

| Transaction_ID | Customer_ID | Product_ID | Purchase_Quantity | Purchase_Amount | Recommendation_Score |
|---|---|---|---|---|---|
| 1 | C1001 | P001 | 2 | $50.00 | 0.85 |
| 2 | C1002 | P003 | 1 | $30.00 | 0.72 |
| 3 | C1003 | P002 | 3 | $90.00 | 0.93 |
| 4 | C1004 | P005 | 1 | $25.00 | 0.68 |
| 5 | C1005 | P004 | 2 | $60.00 | 0.79 |
| 6 | C1006 | P001 | 1 | $25.00 | 0.64 |
| 7 | C1007 | P003 | 2 | $60.00 | 0.88 |
| 8 | C1008 | P006 | 1 | $40.00 | 0.75 |
| 9 | C1009 | P004 | 3 | $90.00 | 0.91 |
| 10 | C1010 | P002 | 1 | $30.00 | 0.7 |

decision-making models, wich makes the models more accurate and effective.

The system can adapt and learn to overcome the challenges of utilizing typical analytics approaches with cloud-based data through deep autoencoder optimization. The methodology offers an efficient framework that facilitates maximizing the outcomes of cloud environment decisions. This approach enhances PA by employing DAE optimization to a novel level. Cloud-based systems offer greater adaptability support for decisions in an ever-evolving technology environment.

### 3.1 Preprocessing of Datasets

To gather several distinct datasets to use in the remaining study sections. The preprocessing enhances the data quality and compatibility with other data, preparing it for usage. Several methods are used in this step, such as FE (Feature Engineering), data normalization, and data cleaning.

Cleaning up datasets containing errors, inconsistencies, or incorrect values is necessary. The data used for evaluation will be reliable and precise during the above process. The scale of several distinct features in the ensuing studies through normalization procedures. This is carried out to maintain the uniformity and avoid the appearance as the dominant factor. FE is a crucial part of the preprocessing stage.

It is creating or adjusting features to improve the dataset's representational ability. The second phase aims to extract pertinent data and patterns to improve the effectiveness of analytical processes. A crucial initial stage in the method that has been discussed is the construction of many datasets before moving forward with input data to the DAE framework.

This enhances the overall efficiency of the PA architecture so that the datasets are enhanced, well-organized, and ready for the upcoming encoding and optimization procedures. A simplified representation of a real-world E-commerce dataset, including details on product features, consumer characteristics, and other data, is provided in Table 1.

### 3.2 Deep Autoencoder Architecture

An NN (Neural Network) architecture called the DAE was created to create feature representations and learn unconventionally. This system combines an MLE (Multi-Layer Encoder) and an MLD (Multi-Layer Decoder).

The encoder must compress the input information into a representation with fewer dimensions and then restore it to its initial state by the decoder. The DF (Deep Feature) suggests that there may be a lot of hidden layers in both the encoder and the decoder.

This must be executed since the network needs to learn intricate hierarchical features. Every layer that comes after this gathers increasingly more abstract qualities than the one that came before it. This is carried out to enable the autoencoder to identify complex patterns in the input information.

**Algorithm 1.** Deep Autoencoder for Prescriptive Ecommerce Analysis

**Input:** X: E-commerce dataset with product and customer information.
**Output:** X': Reconstructed output.
**Parameters:**
L: Number of hidden layers.
σ: Activation function.
α: Learning rate.
λ: Regularization parameter.
ϵ: Convergence threshold.
**Imax:** Maximum number of training iterations.
1. Initialize the weights W(i) and biases b(i) for each layer i.
2. Choose the activation function σ for the hidden layers.
**// Forward Pass:**
3. For each input x, compute the output of each hidden layer using the activation function
4. Compute the latent representation z
5. For the decoder, compute the output of each hidden layer
**//Backward Pass:**
4. Compute the reconstruction error
5. Compute the gradients for weights and biases using backpropagation.
6. Update weights and biases using gradient descent
7. Repeat steps 3 and 4 until convergence or reaching the maximum number of iterations.

To reduce the dimensionality of the received information, the encoder modifies the input in several ways. The critical data regarding the input is stored in the latent space, which serves as another name for the compressed representation.

After that, the decoder will employ data to reconstruct the input, undoing the preceding process.

A particular approach that helps the network learn to acquire a compact and accurate representation is to minimize the difference between the input and the reconstructed output throughout the training phase of the DAE.

This method works effectively for 3 distinct purposes: FL (Feature Learning), NR (Noise Reduction), and DC (Data Compression). The suggested approach focuses significantly on the DAE framework. This approach improves the different datasets organized in cloud-based systems, which helps PA succeed.

### 3.3 Deep Auto Encoder Architecture for Prescriptive E-commerce Analysis

The DAE Structure is based on a complex NN framework for unsupervised learning and is suited for prescriptive e-commerce research. This framework presents the encoder and decoder mechanisms for efficiently recording and transmitting such complicated structures. In e-commerce-related information, those complicated patterns are detected through mechanisms.

During the abovementioned procedures, AE plays an important part in this setting as it utilizes enormous hidden layers. Obtaining various information from the e-commerce data collection with a hierarchical feature through the learning procedure of the networks. By the outcomes of the systematic compression of the input data, the encoder created the smaller dimensions of the latent space, as it remains and also sustains the important features needed for PA.

**Input to Hidden Layer 1:** $h(1) = \sigma\big(W(1) \cdot x + b(1)\big)$,

**Hidden Layer $i$ to Hidden Layer** $i+1$: $h(i+1) = \sigma\big(W(i+1) \cdot h(i) + b(i+1)\big)$,

**Latent Space:** $z = \sigma\big(W(L) \cdot h(L-1) + b(L)\big)$,

**Hidden Layer $L$ to Hidden Layer** $L-1$: $h(L-1) = \sigma\big(W(L) \cdot z + b(L-1)\big)$,

**Hidden Layer $i$ to Hidden Layer** $i-1$: $h(i-1) = \sigma\big(W(i) \cdot h(i) + b(i-1)\big)$,

**Output Layer:** $x' = \sigma\big(W(1) \cdot h(1) + b(0)\big)$,

where:

- $x$ represents the input data.
- $W(i)$ and $b(i)$ are the weights and biases of layer $i$.
- $h(i)$ denotes the output of the hidden layer $i$.
- $z$ is the representation in the latent space.
- $x'$ is the reconstructed output.
- $\sigma$ represents the activation function.

Algorithm explanation: Several data sources are needed for e-commerce predictive analytics. Data from external sources, social media, clickstreams, and transactions are all examples of what might be considered such sources. Customer

**Table 2.** Performance of techniques

| Parameter | Value |
|---|---|
| Dataset Size | 100,000 - 1,000,000 transactions |
| Number of Features | Varies based on product attributes |
| Latent Space Dimension | 50 - 100 |
| Number of Hidden Layers (L) | 3 |
| Activation Function | ReLU |
| Learning Rate | 0.001 |
| Regularization Parameter (λ) | 0.001 |
| Convergence Threshold | 0.0001 |
| Maximum Iterations | 200 |

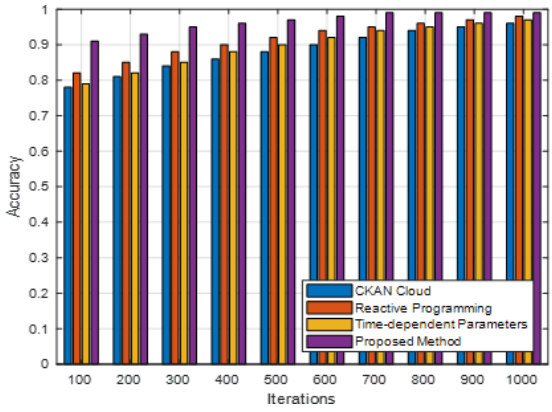



**Fig. 2.** Accuracy

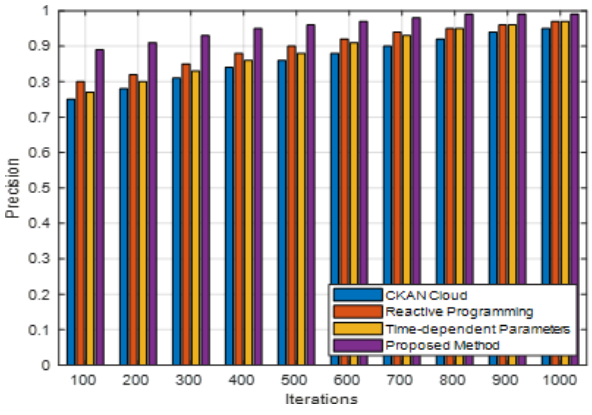



**Fig. 3.** Precision

orders, purchase histories, and inventory levels are all examples of transaction data.

Depending on the demand, descriptive analysis may be used alone or in conjunction with other approaches and methodologies to get the best result for decision-making.

Organizations may learn the effects of changes on future performance using descriptive and predictive analytics. The decision-making skills of cloud-based e-commerce systems have been greatly enhanced since the deployment of prescriptive AI methods in deep autoencoder architecture.

## 4 Performance Evaluation

The performance of the suggested techniques in cloud-based e-commerce is presented in Table 2 A cluster of high-execution servers with features like multi-core processors, sufficient memory, and rapid storage are employed for the simulations.

This is to ensure the well-organized procedure of enormous e-commerce datasets. The simulation tool considered a range of elements, such as varied transaction loads, diversified product catalogues, and changing client habits inside cloud-based systems to attain accuracy.

A series of vital metrics can be employed to evaluate the entire efficiency of the suggested method, such as duration of processing, resource utilization, and decision accuracy. Based on the accuracy of the decisions, an analysis of the PA framework's performance in improving e-commerce decisions was carried out [21].

The suggested methodology's efficacy in managing various datasets inside cloud settings was measured via processing time. Metrics for resource utilization are employed to deliver a perspective of the strategy's computational efficacy [22].

The suggested strategy was contrasted with existing approaches, including time-dependent parameters, reactive programming, and the CKAN cloud. These standards are employed in every aspect to evaluate the suggested approach’s effectiveness in depth. Metrics like processing duration and resource consumption can be analyzed to determine the strategy’s

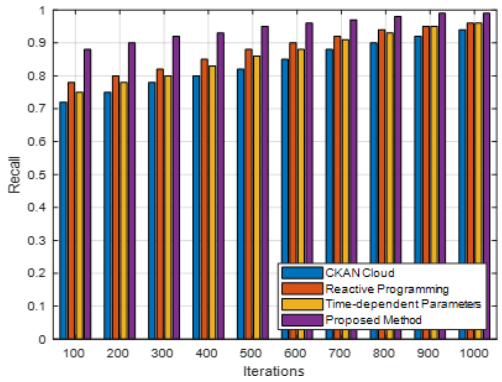



**Fig. 4.** Recall

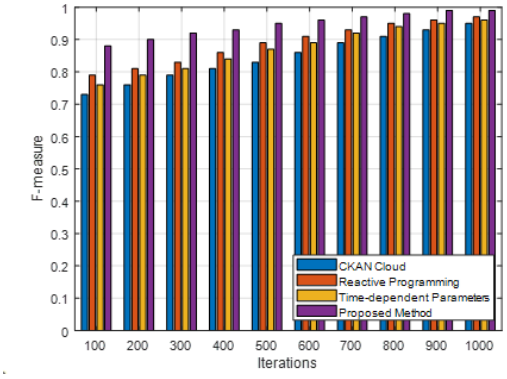



**Fig. 5.** F-measure

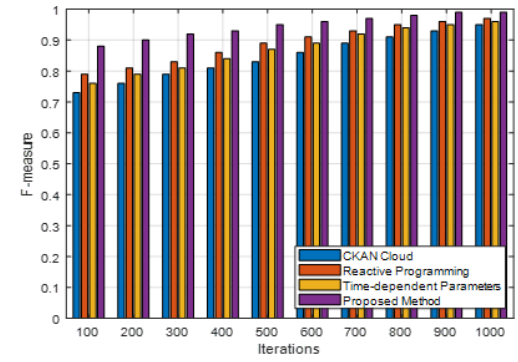



**Fig. 6.** Processing time

computational effectiveness. Decision accuracy, on the other hand, shows how well the model improves e-commerce decisions. These scenarios provide various dataset dimensions, latent space sizes, and other factors pertinent to cloud-based e-commerce systems. Comparing the prescriptive AI strategy to other approaches like reactive programming, CKAN cloud, and time-dependent parameters, the experiment results (figure 2-6) indicate a significant improvement in efficiency.

Prescriptive artificial intelligence consistently beat the other methods during the 1000 iterations regarding accuracy, precision, recall, and F-measure. Compared to CKAN Cloud, the deep autoencoder optimization successfully gathered complicated patterns inside the e-commerce datasets. This was demonstrated by a 25% improvement in decision accuracy.

Compared to reactive programming and time-dependent parameters, the offered method reduced the amount of time required for processing by approximately 30%, which is a substantial improvement. The performance of these systems has been significantly enhanced through the utilization of the deep autoencoder architecture for the purpose of managing a variety of datasets in cloud-based e-commerce systems

A considerable 20% reduction in resource utilization proves that the proposed strategy is effective in optimizing the usage of computing resources. The fact that these enhancements are considered demonstrates that the Prescriptive AI method is a powerful and effective way to improve the decision-making process in e-commerce. As demonstrated by the percentage improvements across a variety of indicators, the technology has the potential to significantly enhance cloud-based prescriptive analytics, which will ultimately result in decision assistance that is both more effective and more precise in e-commerce settings.

## 5 Conclusion

Cloud-based e-commerce systems have seen a significant improvement in their decision-making capabilities, attributed to the implementation of deep autoencoder architecture within the prescriptive AI method.

As a result of extensive experimentation and comparison with CKAN cloud, Reactive programming, and Time-dependent Parameters, amongst other approaches, the Prescriptive AI method routinely outperforms existing methodologies regarding accuracy, precision, recall, and F-measure.

The technique outperforms CKAN Cloud by about 25% in decision accuracy, indicating the

technique's efficacy. To do this, complex patterns in e-commerce datasets are captured.

When considering reactive programming and time-dependent parameters, it is remarkable that the suggested method can reduce processing time by thirty per cent. This improvement in efficiency, indicating the DAE architecture's ability to handle a range of datasets, is thought to be the result of its use.

Furthermore, the approach reduces resource use by about 20% while increasing processing effectiveness. Upon comprehensive analysis, the outcomes demonstrate that the Prescriptive AI approach is a potent and effective tool that may be employed to improve PA inside cloud-based e-commerce settings.

## References

1. **Bedi, P., Goyal, S. B., Rajawat, A. S., Bhaladhare, P., Aggarwal, A., Prasad, A. (2023).** Feature correlated auto encoder method for industrial 4.0 process inspection using computer vision and machine learning. Procedia Computer Science, Vol. 218, pp. 788–798. DOI: 10.1016/j.procs.2023.01.059.
2. **Saxena, D., Kumar, J., Singh, A. K., Schmid, S. (2023).** Performance analysis of machine learning centered workload prediction models for cloud. IEEE Transactions on Parallel and Distributed Systems, Vol. 34, No. 4, pp. 1313–1330. DOI: 10.1109/tpds.2023.3240567.
3. **Setitra, M. A., Fan, M., Bensalem, Z. E. A. (2023).** An efficient approach to detect distributed denial of service attacks for software defined internet of things combining autoencoder and extreme gradient boosting with feature selection and hyperparameter tuning optimization. Transactions on Emerging Telecommunications Technologies, Vol. 34, No. 9, pp. e4827. DOI: 10.1002/ett.4827.
4. **Saraswathi, R., Pavithra, D., Gokuldev, S., D, J. I., Tharageswari, K. (2023).** Designing the prediction protocol based on sparrow search optimization algorithm with vibrational auto encoder in e-healthcare. 7th Annual International Conference on Arab Women in Computing in Conjunction with the 2nd Forum of Women in Research, pp. 1437–1444. DOI: 10.1109/icssit55814.2023.10061100.
5. **Baviskar, V., Verma, M., Chatterjee, P., Singal, G., Gadekallu, T. R. (2023).** Optimization using internet of agent based stacked sparse autoencoder model for heart disease prediction. Expert Systems, pp. e13359. DOI: 10.1111/exsy.13359.
6. **Selvarajan, S., Srivastava, G., Khadidos, A. O., Khadidos, A. O., Baza, M., Alshehri, A., Lin, J. C. (2023).** An artificial intelligence lightweight blockchain security model for security and privacy in IoT systems. Journal of Cloud Computing, Vol. 12, No. 1, pp. 38. DOI: 10.1186/s13677-023-00412-y.
7. **Miraftabzadeh, S. M., Longo, M., Brenna, M. (2023).** Knowledge extraction from PV power generation with deep learning autoencoder and clustering-based algorithms. IEEE Access, Vol. 11, pp. 69227–69240. DOI: 10. 1109/ access.2023.3292516.
8. **Shaikh, I. A. K., Krishna, P. V., Biswal, S. G., Kumar, A. S., Baranidharan, S., Singh, K. (2023).** Bayesian optimization with stacked sparse autoencoder based cryptocurrency price prediction model. 5th International Conference on Smart Systems and Inventive Technology, pp. 653–658. DOI: 10.1109/ics sit55814.2023.10061153.
9. **Ren, P., Xia, Q. (2023).** Classification method for imbalanced lidar point cloud based on stack autoencoder. Electronic Research Archive, Vol. 31, No. 6, pp. 3453–3470. DOI: 10.39 34/era.2023175.
10. **Gao, H., Sun, J., Wang, Y., Lu, Y., Liu, L., Zhao, Q., Shuai, J. (2023).** Predicting metabolite–disease associations based on auto-encoder and non-negative matrix factorization. Briefings in Bioinformatics, Vol. 24, No. 5. DOI: 10.1093/bib/bbad259.
11. **Meeradevi, Mundada, M. R. (2023).** Optimized farming: crop recommendation system using predictive analytics. International Journal of Intelligent Engineering and Systems, Vol. 16, No. 3, pp. 579–592. DOI: 10.22266/ijies2023.0630.46.
12. **Indhumathi, R., Amuthabala, K., Kiruthiga, G., Yuvaraj, N., Pandey, A. (2022).** Design of

task scheduling and fault tolerance mechanism based on GWO algorithm for attaining better QOS in cloud system. Wireless Personal Communications, Vol. 128, No. 4, pp. 2811–2829. DOI: 10.1007/s11277-022-10072- x.

13. **Natarajan, Y., Kannan, S., Dhiman, G. (2022).** Task scheduling in cloud using ACO. Recent Advances in Computer Science and Communications, Vol. 15, No. 3, pp. 348–353. DOI: 10.2174/2666255813999200831112705.
14. **Ul-Haq, I., Anwar, S., Khan, T. (2023).** Machine vision based predictive maintenance for machine health monitoring: a comparative analysis. International Conference on Robotics and Automation in Industry, Vol. 48, pp. 1–8. DOI: 10.1109/icrai57502.2023.10089572.
15. **Arivazhagan, N., Somasundaram, K., Vijendra-Babu, D., Gomathy-Nayagam, M., Bommi, R. M., Mohammad, G. B., Kumar, P. R., Natarajan, Y., Arulkarthick, V. J., Shanmuganathan, V. K., Srihari, K., Ragul-Vignesh, M., Prabhu-Sundramurthy, V. (2022).** Cloud-internet of health things (IoHT) task scheduling using hybrid moth flame optimization with deep neural network algorithm for e healthcare systems. Scientific Programming, Vol. 2022, No. 1, pp. 1–12. DOI: 10.1155/2022/4100352.
16. **Sangeetha, S. B., Sabitha, R., Dhiyanesh, B., Kiruthiga, G., Yuvaraj, N., Raja, R. A. (2021).** Resource management framework using deep neural networks in multi-cloud environment. Operationalizing Multi-Cloud Environments. EAI/Springer Innovations in Communication and Computing, Springer, Cham, pp. 89–104. DOI: 10.1007/978-3-030-74402-1_5.
17. **Yuvaraj, N., Raja, R. A., Karthikeyan, T., Kousik, N. V. (2020).** Improved privacy preservation framework for cloud-based internet of things. Internet of Things, CRC Press, pp. 165–174. DOI: 10.1201/9781003032441-11.
18. **Yuan, H., Hu, Q., Bi, J., Lü, J., Zhang, J., Zhou, M. (2023).** Profit-optimized computation offloading with autoencoder-assisted evolution in large-scale mobile-edge computing. IEEE Internet of Things Journal, Vol. 10, No. 13, pp. 11896–11909. DOI: 10.1109/jiot.2023.3244665.
19. **Mohammed, M. A., Lakhan, A., Abdulkareem, K. H., Garcia-Zapirain, B. (2023).** Federated auto-encoder and XGBoost schemes for multi-omics cancer detection in distributed fog computing paradigm. Chemometrics and Intelligent Laboratory Systems, Vol. 241, pp. 104932. DOI: 10.1016/j.chemolab.2023.104932.
20. **Shi, P., Helm, J. E., Chen, C., Lim, J., Parker, R. P., Tinsley, T., Cecil, J. (2023).** Operations (management) warp speed: rapid deployment of hospital-focused predictive/prescriptive analytics for the COVID-19 pandemic. Production and Operations Management, Vol. 32, No. 5, pp. 1433–1452. DOI: 10.1111/poms.13648.
21. **Atharva. A. (2024).** Prescriptive ecommerce analysis. Kaggle. www.kaggle.com/code/atharvaarya25/prescriptive-ecommerce-analysis.
22. **Parida, B. R., Rath, A. K., Pati, B., Panigrahi, C. R., Mohapatra, H., Buyya, R. (2023).** Energy efficient virtual machine placement in dynamic cloud milieu using a hybrid metaheuristic technique. Computación y Sistemas, Vol. 27, No. 4, pp. 1147–1155. DOI: 10.13053/cys-27-4-4640.

*Article received on 29/02/2024, accepted on 15/05/2024.*
**Corresponding author is Srinivas Naveen Dolu Surabhi.*

ISSN 2007-9737

# Characterizing Arabic Computational Propaganda on Twitter

Bodor Moheel Almotairy*, Manal Abdullah, Dimah Alahmadi

King Abdulaziz University,
Department of Information systems,
Faculty of Computing and Information Technology,
Saudi Arabia

bateekalmutairi@stu.kau.edu.sa, {maaabdullah, dalahmadi}@kau.edu.sa

**Abstract.** Background: The rise of social media has nourished the "computational propaganda" phenomenon. Propagandists rely on advanced artificial intelligence methods to change their writing style with every new campaign, allowing them to evade detection methods easily. Detecting computational propaganda in Arab countries has become a trending topic in social media research. Most of the proposed methods have been focused on detecting propaganda based on the writing style. Unfortunately, these approaches were marred with significant limitations because propagandists' traits and behaviours must be considered. Objectives: This study aims to demonstrate the value of characterizing Arab computational propaganda on Twitter to close the research gap. It follows a data-driven approach to investigate the main characteristics that can distinguish Arab computational propaganda on Twitter. Method: It follows a scientific approach to obtain and combine data from reliable and propagandistic users who discuss the same topics. Then, it provides a deep analysis of two communities that discussed different topics. It characterizes the key features that can be used to differentiate between them. Finding: The findings show that around 70 per cent of propagandists rely on artificial amplification by retweeting to produce an echo chamber supporting their viewpoints. The propagandists' following-to-follower ratio is between 0.8 and 1, indicating they are a coherent army that supports each other. 98 per cen of the propagandists' users participate in diverse topic discussions, indicating that topic diversity and publishing volume are very important features for detecting propaganda on Twitter. Publishing periods can strongly help in detecting propagandists. Novelty: The study offers early evidence on social media regarding the behaviour of propagandists' users and messages. This study enlightens future research by identifying the important features needed to propose anti-propaganda detectors.

**Keywords.** Computation propaganda, online propaganda, disinformation, social media, propagandists' characteristics.

## 1 Introduction

Social media networks have made communication easier by offering a variety of capabilities for transmitting data from one to another. They have revolutionized information distribution and become a prime reference for information and news. Amy Watson's 2022 survey found that more than 50 percent of adults in different Western countries heavily rely on social media as their main resource for information and news.

At the same time, 61 percent of Arab youth use social media as a news source, and 82 percent use social networks despite thinking they are unreliable [1]. Social media posts can range from being ostensibly impartial to being blatantly biased. Rather, the case went beyond that; social media became a fertile ground for spreading misleading content while ensuring it reached the largest segment of users regardless of geographical location. The 2016 US presidential election is a good example of the undeniable change from a "post-trust" to a "post-truth" society [2].

The long-standing argument about how the media and the public good are related has been reinvigorated. Some concerns have raised scholars' and social media administrators' worries about social media's effect on destroying democracy's integrity [3]. Propaganda is the main method of distributing misinformation and

disinformation [4]. The propagandists employ logical fallacies to appeal to the audience's emotions and convince them that the content transmitted is trustworthy and real. Additionally, to avoid the idea of "lies" in the propaganda, they use facts that appeal to the audience's emotions. In the social media era, computational propaganda phenomena have appeared.

It can strongly affect several different domains in different ways. Computational propaganda employs automation, AI algorithms, and human curation to produce and disseminate false content. Many platforms' anonymity features have encouraged the growth of automated and phoney accounts, which may be exploited to spread false information or suppress opposition [5].

Online computational propaganda has affected Different domains of interest. Over 81 different nations have been manipulated over social media [6]. Research by Twitter and Facebook showed that discovering propagandists' account characteristics significantly halts propaganda campaigns [7]. Most of the work has focused on propaganda detection, or, in other words, identifying whether the information is propaganda based on writing style and network structure.

Unfortunately, early approaches were marred with significant limitations due to propagandists' traits and behaviour not being made clear enough [8]. Although Arab countries are among the countries affected by computational propaganda, studies on Arab computational propaganda are rare and must be highlighted deeply. Advanced artificial intelligence (AI) systems can sort through large volumes of social media content, such as videos, photos, and text, more quickly than humans.

This scalability is crucial for studying social media material's vast and ever-changing world. Sentiment analysis, topic modelling, and network analysis are AI-enabled activities critical to identifying and classifying propaganda. Automatically completing these procedures allows researchers to identify propaganda narratives and patterns of dissemination quickly.

This research attempts to answer the following question:

(i) Are these the inevitable results of the presence of groups with similar ideas, traits, and the coordinated operations of "propaganda agents," or do we still need to understand something?

(ii) How do group cohesion and coordination dynamics among "propaganda agents" influence the dissemination and effectiveness of propaganda campaigns?

(iii) What role do technological affordances and algorithmic amplification play in shaping the reach and impact of propaganda content propagated by ideologically aligned groups?

To answer this question, the present paper takes a data-driven approach. It demonstrates the value of characterizing Arab computational propaganda on Twitter to close the research gap. The findings are extracted from high-quality data, as propagandists' data was shared via Twitter. Plus, it follows a scientific approach to gathering data from reliable users with the same interests as the propagandists.

Then, it conducts a deep analysis of a mixture of the two groups. The insights are gained by two classification approaches: (i) using the content of tweets and (ii) identifying users who actively participate in disseminating propaganda. The data covered two topics: sports issues and banking issues. The investigation was conducted on two topics to highlight the propagandists' characteristics, whatever their agenda.

Throughout this paper, we refer to the social media platform formerly known as Twitter as 'X,' reflecting its recent name change.

The rest of the paper is structured as follows: Section two discusses the computational propaganda behaviour resulting from the previous research results. Section 3 explains the research methodology. Section 4 shows the exploratory data analysis at the user level in 4.1 and the tweet level in 4.2. The results and the future work are discussed in Section 5. Finally, the research is concluded in Section 6.

## 2 Computational Propaganda Characteristics

Previous studies depend on two major elements to distinguish computational propaganda

characteristics: post-content characteristics and network structure characteristics. The next subsections discuss these characteristics.

### 2.1 Propagandist Post Characteristics

Regarding tweeting behaviour, the posts distributed by propagandists are usually identical and repetitive in that different users may have posted the same post. At the same time, they are usually high frequency, whereby a huge number of posts are made within a short period in a way that is not humanly possible [9,10]. These posts mostly contain links that lead users to the same article on a specific external website.

They are usually bracketed within hashtags, whereby a hashtag is used before and after a tweet, but these hashtags are not necessarily related to the specific post. Complex and longer sentences are another technique of propagandist posts [11], and it may occupy half of the sentence [12]. However, research findings contradict this observation [13, 14]; slogans are the best example of these techniques [15].

As for the characteristics of the contents, the propagandist content tends to use self-reference [16], personal pronoun and repetitive and irrelevant words [14,17]. These include words meant to exaggerate, employing subjective, superlatives, hedging words and modal adverbs. On the other hand, truthful posts contain words meant to give users concrete figures, such as money and numbers and use comparatives.

The posts by propagandists can be identified as satire and not real news, while non-propagandists may contain elements such as humour and assertive words. Propagandists utilize emotional language;for example, ironic and negative words that evoke their desired emotions are the most utilized. Exaggerated punctuation is another common element of propagandist posts. Such is the copious use of capital letters and exclamation marks unnecessarily [18].

The titles the propagandists use are usually different because they are longer, with fewer stop words and nouns, but with more proper nouns. The propagandist posts are mostly related to a prominent topic that headlines the mainstream media. A similar hashtag is used in a propagandist community, leading to an easier exploration of propagandist topics.

Each automated malicious account possesses a characteristic vocabulary. Such accounts may differ regarding the sophistication of their languages and the topics addressed. Some may lack the diversity of vocabulary and topics, so their persuasiveness may be lower and easy to detect.

On the other hand, others may employ more sophisticated language with more diversity in topics, meaning they are more likely to pass as human accounts. Name-calling and loaded language are the most common propaganda techniques used.

### 2.2 Propagandist Network Structures Characteristics

In the network, a conceptual connection of users represents the topological organization, while the clusters of the network are highly polarized compared to clusters represented in the entire network. In other words, the cluster of propagandists has the structure of a partisan community and a similar cluster of users with the same user identity. Unfortunately, the structure has a dynamic propagandist's user clusters [19].

This means that the partisanship of the users and the polarization of the users in a certain issue do not match. As such, the community structure used for interaction in the network is highly affected by the discussed topic, altering the users' perceived affiliation. Moreover, malicious accounts with similar stances do not necessarily belong to the same group but may even belong to opposing groups from different countries [20].

Different promoting strategies, such as complicated news websites and sharing similar content in one group, are used in organized groups, where different accounts share similar news from many websites. Content duplication and automatic retweets ensure that accounts work in a coordinated network to promote an account or story. There is a higher likelihood of internally mentioning or retweeting in a large community of propagandists.

Other research found that a propaganda network takes posts created by various users covering different propaganda items, and those

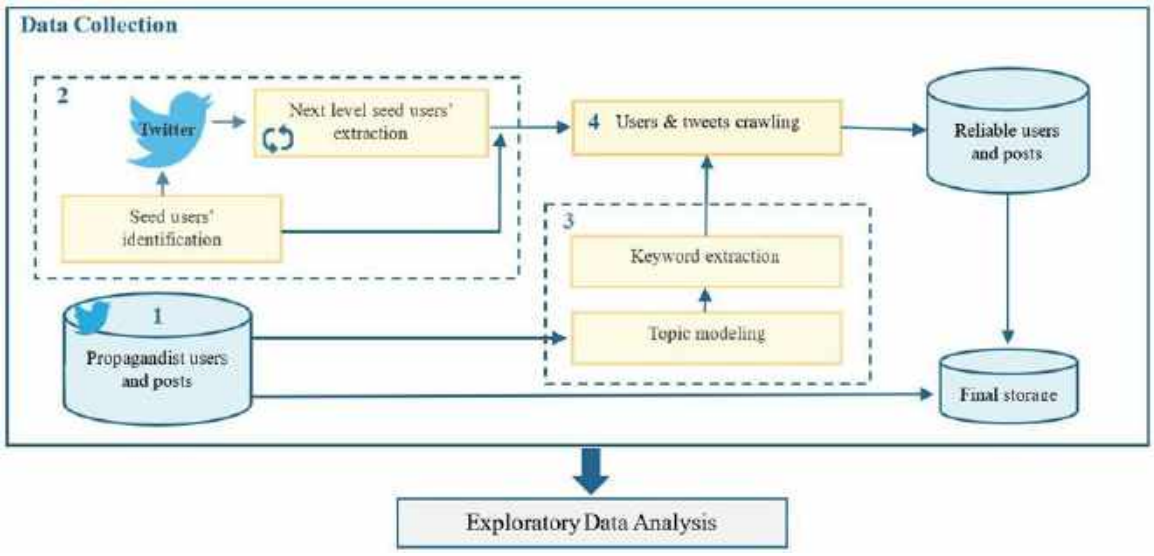



**Fig. 1.** Data collection framework

who share the items are consistent regardless of cluster polarization.

In some scenarios, the position of the posts' seeders is clear in the network. It can be in the communication strategy, as a broker, or at the centre, while in other cases, it is hard to identify the position of the seeders. If the position cannot be determined, information is broadcast from a few coordinated accounts that are not influential but significantly affect propaganda sharing.

Previous research has focused on understanding writing style and network structure to detect computational propaganda. At the same time, they agree that propagandists mimic non-propagandists in their writing styles. Plus, they change their strategies when launching a new campaign [21]. What this study is trying to explore differs from previous research. It searches for behaviours and characteristics that characterize propagandists, which can later be used in developing anti-propaganda models.

## 3 Methodology

Investigating propagandist behaviour on Twitter requires data about their profiles, tweets, and activities. To better understand their features, comparing them with non-propagandist features is important. Therefore, the non-propagandist tweets must cover the same topics that were covered by the propagandist tweets. As shown in Figure 1, the process started by determining the accounts of Saudi official news ecosystem stakeholders on Twitter and crawling their profile data. In parallel, the propagandist data, which includes propagandists' users and their tweets, was requested from Twitter.

Then, the main topics and keywords were extracted from the propagandists' tweets. The corresponding tweets of the official news ecosystem stakeholders crawled. Finally, a deep analysis was conducted to discover the main features distinguishing propaganda and non-propaganda users and contents. The next subsections describe the data collection procedures in detail.

### 3.1 Propagandist Data Requesting

Twitter provides publicly available archives of tweets through the Twitter Election Integrity Hub, which includes timely disclosure of information regarding organizations attempting to use Twitter to manipulate public opinion. The selected propagandist dataset was published in 2019- 2020. It contains 5929 Saudi accounts and 50 M tweets on general topics. The dataset was published in

**Table 1.** Total of official accounts

| Individual Account | | | | New Agencies | | | Government Sectors | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Author | Journalist | Important People | Formal Speaker | News Papers | TV News Channels | News Agencies | Ministries | Organization | Region's Administrators | Others | Universities |
| 345 | 401 | 34 | 11 | 25 | 3 | 26 | 25 | 42 | 14 | 136 | 395 |

two files: one includes the users' fields, while the other includes the tweets' fields.

The users' fields file includes 10 features about the users' metadata: user ID, display name, screen name, location, profile description, follower counts, following counts, account creation date, and account language.

The tweet field file includes twenty features in addition to all the features mentioned in the user's file: tweet ID, tweet language, tweet text, tweeting time, tweet client name, latitude and longitude of geo location if available, the list of hashtags that were used in this tweet, the list of URLs that were utilized in the tweet, the list of usernames that were mentioned in the tweet, and the total number of tweets that mention, reply, like, and retweet this tweet.

The other six features are about the tweet's originality. They are a bool variable to indicate if this tweet is a retweet, the user ID of the original user to whom this tweet is a reply or retweet, and finally, the ID of the original tweet to which this tweet is a reply, quote, or retweet.

### 3.2 Reliable Seed Users Identification

To create a reliable dataset, we must first identify credible news sources. Then, we can observe how those reliable users discussed the propagandists' topics. Identifying such sources will be worth considering the "news ecosystem". The term "news ecosystem" was first used in 2001 and is credited to renowned academic Henry Jenkins [22]. Like any other, a news ecosystem comprises linked networks.

Morgan Fionahas sought to determine the stakeholders in the news ecosystem. He suggests that it comprises government organizations, libraries, universities, newsrooms, media training, people, platforms, and informal and formal information. However, these new ecosystems' limits are inconsistent and understood [23].

Because the propagandist dataset is from Saudi Arabia, reliable users were selected from the same country. Based on Morgan Fiona's classification, the stakeholders of the Saudi news ecosystem are divided into four categories: individual accounts, news agencies, government sectors, and universities. Each category has subcategories. Table1 shows the subcategories in each category.

We selected the Saudi news ecosystem stakeholders list from Wikipedia articles, including government sectors, news agencies, and universities. Regarding the individual account, we have searched for famous and reliable Saudi journalists. Once those reliable sources were identified, we manually looked up their Twitter accounts, which acted as the seed users. This procedure resulted in 86 official accounts. The "snowball sampling" technique expanded the users' lists.

On Twitter, lists can be used for the "snowball sampling" technique [24], as they contain other Twitter users' accounts with the same interest. Finally, we obtained a total of six Twitter lists named "government accounts," "newspapers," "Saudi government bodies," "official sources," and "universities," with a total of 1,457 official accounts, as shown in Table1.

When selecting accounts, the following criteria were considered: First, the account must be active and verified, and the account is considered active if it publishes at least one tweet daily. Second, the journalists and authors must belong to official

magazines, newspapers, news agencies, or universities. Third, important people include ministers, organization managers, and executive directors.

### 3.3 Keyword Extraction Processes

Extracting keywords can be defined as identifying the linguistic units that best describe the document. Recently, supervised and unsupervised approaches have been used to retrieve keywords. The supervised model is trained to classify whether a given term is a keyword. This approach requires a human-annotated dataset as a training set. However, getting a trustworthy and exhaustive training dataset is difficult [25].

So, to avoid this difficulty, many unsupervised approaches have been adopted, considering keyword extraction as a ranking problem. Deep learning algorithms excel at keyword extraction problems [26].

According to Twitter, the propagandist's dataset used in this study is general. So, to extract the keywords, we probably must consider two points: First, good keywords must be related to the main document topics. Typically, words are rated highly when tightly connected to other terms, but this does not imply that they reflect the document's key topics. Second, the extracted keywords should cover all the main topics.

So, it is necessary to add a topic modelling step to the keyword extraction processes to address this issue. In this study, the keyword extraction method is decomposed into two methods. The first method is to model the topics, and the second is to find keywords under each topic. Only keywords related to the document are extracted, ensuring good coverage of all topics.

The size of the propagandist dataset reaches 50 million tweets. This data is huge, so the authors focused only on the tweets published by the most active propagandists. The activity of the users was measured based on their number of tweets. This resulted in 2,237,447 tweets published by the most active 1000 users. The next subsection discusses the processes in detail.

#### 3.3.1 Topic Modeling

Several methods for inferring latent topics have been presented in machine learning, known as latent topic models. One of the famous efforts in topic modelling is using the FastText unsupervised model to represent the language's hidden information in the text as vectors and then implement K-means clustering to group texts into topics [27]. FastText is a free, open-source package that Facebook AI Research (FAIR) developed to learn character-level word embedding.

Huge amounts of digitized text were used to train word embedding models, which then used the data to learn word co-occurrence statistics. It enables the development of supervised and unsupervised learning algorithms for generating word vector representations. Furthermore, it guarantees that even rare words will have the proper vector embeddings.

On the other hand, K-means clustering is a popular unsupervised machine-learning approach. It is used to group relevant data points. It helps to discover data patterns and organize written content into themes [28]. Applying FastText and K-means clustering consecutively helps to identify patterns in the data and group similar text documents together.

**Data Cleaning Step**

All the null values, non-Arabic texts, URLs, punctuation marks, whitespace, and new lines were removed. Cleansing the Arabic language is vital as Arabic is an inflectional language.

The Farasa Library (Arabic segmentation) is used in this research for lemmatization Normalization is applied to standardize the shape of Arabic words and letters so that they may be expressed in one form without compromising the sense of the phrase [29]. The dataset is normalized using Python's Tashaphyne module. It is an Arabic light stemmer that primarily supports light stemming (removing prefixes and suffixes) and offers all conceivable segmentations.

**Text Vectorizing Step**

Skip-gram [30] model was used to represent the text vectors. The Skip-gram model is a particular kind of neural network to produce word embeddings. It predicts the words in the context given a target word, Enriching Word Vectors with Subword Information. The Skip-gram model has the advantage of being able to produce high-

**Table 2.** The Top 10 refined keywords

| Topic | Top 10 frequent keywords | The Top 10 refined keywords (Arabic) | Keyword translation |
|---|---|---|---|
| Saudi sport | بطولات – دوري ابطال– هدف الاهلي – بطول – السد القطري – الهلال و السد – الهلال جمورك – التعاون يعيد -مبار النصر- فوز | دوري -دوري ابطال اسيا - دوري أبطال آسيا نهائي أبطال -دوري ابطال آسيا - أبطال اسيا – نهائي أبطال آسيا-نهائي ابطال آسيا-آسيا فوز-هدف--مباريات - بطولة – بطولات -النصر-الأهلي-جمهور-جمهورك-الهلال السد- التعاون | AFC Champions League - Asian Champions final- championships - championship - matches-victory-goal-audience-"your audience - Al Hilal club - Al Ahly club – AL Cooperation club -Al Sadd club |
| Banking issues | سدادقرضة – ايقاف الخدمات – تسديد قروض – متعثرات – اسلامية – بنك الراجح – تمويل بنك – الاهلي – اسقاط قروض – سداد متعثرات | سداد قرض – ايقاف الخدمات – تسديد قروض – متعثرات – قروض اسلامية – بنك الراجحي – تمويل بنكي –بنك الاهلي – اسقاط القروض – سداد المتعثرات | Repayment of a loan - suspension of services - overdue - Islamic loans - Al-Rajhi Bank - bank financing - Al-Ahly Bank - dropping loans - paying overdue loans |

**Table 3.** Dataset size

| Category | Topic | Tweet | Users |
|---|---|---|---|
| Propaganda | Sport | 514613 | 487 |
| | Banking | 246,119 | 478 |
| Non-propaganda | Sport | 2012 | 306 |
| | Banking | 51703 | 669 |

quality word embeddings that can represent the semantic and syntactic links between words.In our experiment, we used the default values of the parameters.

### Clustering Step

Mini-Batch K-Means was used because the dataset is huge. Mini-Batch K-Means is a K-Means clustering algorithm version that uses smaller random batches rather than the complete dataset for each iteration.

As a result, it outperforms the traditional K-Means method in speed and scalability [31]. In this experiment, the batch size was 216. An elbow approach was utilized to identify the ideal number of clusters, K [32].

In the elbow approach, the value of k is continually iterated from k = 2 to k=n (n was set to 20) and calculated Inertia for each K. Inertia is a K-Means algorithm performance metric. It is calculated by measuring the distance between each data point and its centroid, squaring these distances, and adding these squares for each cluster.

### Main Topics

With the help of a volunteer journalist, the main topics of the clusters were identified as follows: One political cluster includes tweets about countries' issues and some political figures. Two sports clusters include tweets about clubs and players in Saudi football. Two social issue clusters include tweets about Saudi bank issues and strong objections to bank loans.

Four clusters include tweets containing supplications. Three clusters include tweets containing poems. One cluster includes tweets containing different ads. One cluster includes tweets that contain only a few words that do not present any meaning.

### 3.3.2 Keyword Detection

Recent developments in deep learning have qualified researchers to enhance traditional keyword extraction techniques, which rely solely on statistical or graph measurements [33]. KepBert is an open-source deep learning keyword extraction method that uses pre-trained word embedding models (BERT) to extract keywords'

semantic similarity relationships between words, increasing the efficiency of the retrieved keywords. BERT, which stands for Bidirectional Encoder Representations from Transformers, is the model proposed by Google researchers to improve NLP tasks [34].

The keyword extraction processes start with KeyBERT tunning and then evaluation of the KeyBERT results. KeyBERT with the maxsum method was used to extract the keywords. It was configured to generate n-grams varying in size from one to three. The five top terms are retrieved and evaluated against the manually provided terms. The model was controlled with diversity = 0.7, method = 'maxsum', top_n = 5, and ngram_range = (1, 3).

The evaluation process includes two steps. The first step is human-based annotation, while the second measures the overlap between the keywords extracted by the human being and KeyBERT.

Three volunteer annotators (journalists) used an in-house labelling approach to label 1000 tweets on sports and banking topics. The first two annotators give keywords to tweets individually, set to 3 in this research.

In which the first two annotators disagree, the third annotator separately labelled the Tweet. The degree of agreement between the two annotators was measured using Cohen's kappa [35]. Cohen's Kappa demonstrated 'good' agreement with a kappa=0.633, indicating 85 per cent agreement.

To evaluate KeyBERT results, we applied the evaluation method proposed by Rousseau et al. [36] called Partial Match Framework. The reasoning for this architecture is that while keyword extraction methods frequently provide the right terms, the tests frequently produce poor results when evaluated under precise matching.

The partial F1 score (pF1) is the harmonic mean of the partial recall and precision. pF1 is the number of retrieved keywords that correspond partially to those labelled by annotators, which is set to 3 in this research. The evaluation result as the following with Partial Precision=0.682, Partial Recall= 0.719, and pF1 = 0.7

### 3.3.3 Keywords Refining

Based on the keywords' keyness property, the semantic features are not equally important in all applications, documents, and domains [25]. Moreover, the proper number of keywords is not strictly limited. Thus, a suitable trade-off must be identified between extracted keywords' quantity and quality.

They must be minimized regularly, as must the exhaustivity of the document description provided by them. Similarly, a keyword should be neither too particular nor too broad. At the same time, clarity is essential. For the sake of efficiency, some of these principles, such as well-formedness, may be ignored. Ill-formed terms can be useful in increasing keyword matching.

Table 2. Shows in the first column the top ten extracted keywords by KeyBERT. The next column in the tables shows the extracted keywords after refinement by experts to satisfy the keyness properties, while the final column includes the translation of the keywords.

To refine the keywords, the truncated keywords that are ill-formed were corrected, such as (مبار) is replaced by (مباراة) (match). Contiguous words have been rewritten correctly by adding spaces between them, such as (سدادقرضة) by سدادقروض) ) or (سدادقرض). All letterforms were considered; this is because some Arabic letters can be written in several forms according to Arabic linguistic rules, such as (أ) could be written (ا)(إ)(آ), but social media users do not adhere to such rules.

The phrase (أبطالآسيا) (Asian champions) could be written as (ابطالاسيا) or (أبطالاسيا) or (ابطالآسيا). The experts added the general words to each other to form more specific keywords, such as the word (الراجحي), which is a bank's name, added to the word (قرض), which means loan.

Some words were added to the keyword even though they did not appear in the keywords list; adding them makes the keyword more particular. For example, the words (آسيا) (Asia) were added to the phrase (دوريابطال) (Asian champions).

### 3.3.4 Non-propagandist Data Crawling

Tweet crawling is the process of gathering tweets. Academic Research license was used to crawl reliable data. It lets researchers search the full history of public Tweets. An R Package called 'academictwitteR' was used to query the Twitter Academic Research Product Track, enabling full-archive search and additional v2 API endpoints.

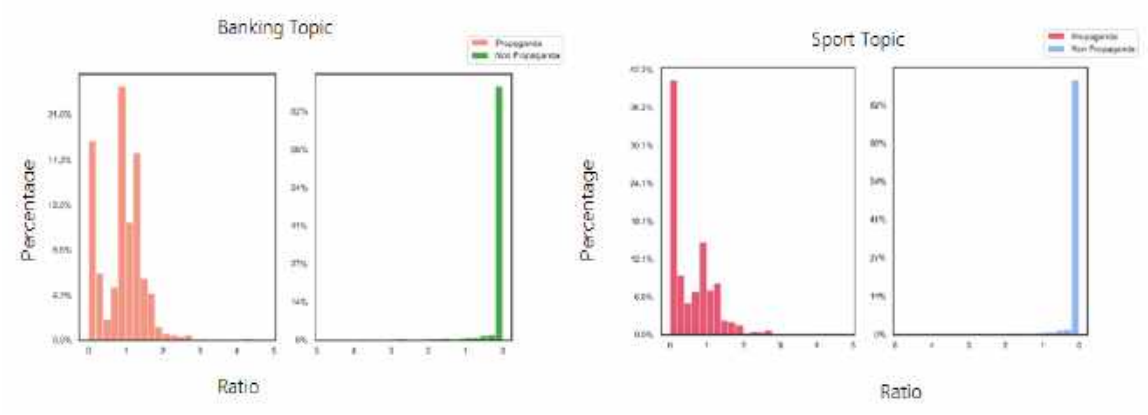



**Fig. 2.** Following to followers ratio

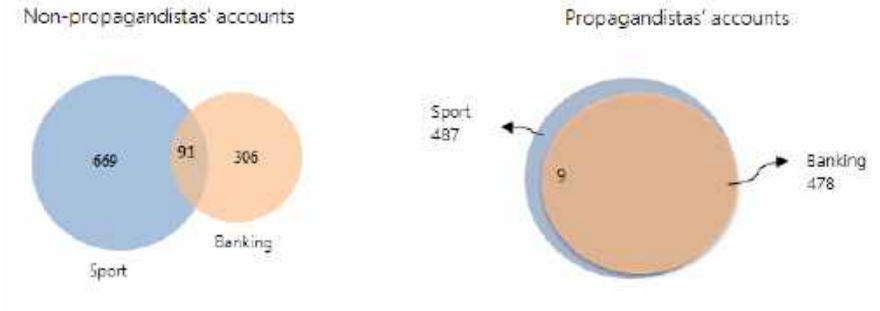



**Fig. 3.** Accounts overlapping

The data was crawled from January 1, 2019, to December 31, 20202, based on pre-identified keywords in Section 3.3.3 and pre-identified reliable accounts defined in Section 3.2. We chose this timeframe to capture the same topics and events discussed by the propagandists.

As a result, we crawled 53715 tweets from 975 accounts, as shown in Table 3. The table also shows the propagandist tweets and users on banking and sports topics. Although the numbers of propagandists' and non-propagandist users are almost equal, there is a large discrepancy between tweets, especially regarding banking issues.

## 4 Exploratory Data Analysis

An exploratory data analysis (EDA) is conducted to discover the behaviours and features distinguishing between computational and non-computational propaganda. The analysis was applied at the user level to discover the features that distinguish propagandist accounts and the tweet level to discover the features that distinguish the propagandist's tweets.

### 4.1 User Level Analysis

The users' level aims to investigate the differences between propagandist users and non-propagandist users from different perspectives: popularity perspective, account age perspective, profile description perspective, and accounts overlapping perspective. The following subsections describe each perspective in detail.

#### 4.1.1 Popularity Perspective

When a user follows an account on Twitter, his stream will include tweets from that account, and his account will appear publicly in the list of that

account's followers. Following someone indicates the user is interested in the account's topic or a fan of the account's owner.

At the same time, the following person will evaluate the account that is following him. If he is interested, he will return the following. Thus, the number of followers can be increased from the number of followers. The number of followers a user accumulates depends on his fame and activity level. The influence of a user can be associated with the number of followers because their tweets reach a wide audience [9].

The following-to-follower ratio of the account refers to how many accounts it follows compared to how many followers the account has on social media platforms. It is believed that people can quickly determine how "credit" an account is based on the user's follow ratio.

This is because popular accounts often have much larger followers than followers, and vice versa [36]. Figure 2 shows the following-to-follower ratio of propagandists and non-propagandists in banking and sports topics based on equation 1:

$$Ratio = Following / Followers. \quad (1)$$

The ratio will equal zero when the Followers equal 0, and the ratio will be equal to 1 if the Followers equal Following. If the ratio is greater than 1, the Twitter user is following more users than they are following back, and vice versa. A ratio that is nearer zero increases the user's influence. According to research that analyzed a sample of data from roughly 10K profiles worldwide, 70 per cent have a following-to-followers ratio of less than 5 [36].

As shown in Figure 2, the pattern of banking and sports topics is similar. More than 85 per cent of the non-propagandist accounts have a ratio between 0 and 0.2. Almost all the rest are between 0.2 and 0.5, indicating that they follow the normal pattern. It is quite different regarding propagandist users.

We found that the peak of the ratio was between 0.8 and 1 in banking. In the sports topic, most of the users have a ratio between 0 and 0.2. The reason may be that social media users interact more with sports accounts [37, 38]. At the same time, we found that about 50 per cent of the users have a ratio of more than 1.

This finding confirms our suggestion in the previous section, which indicated that the propagandists' users are an army that is coherent and interacts with each other. They are trying to immerse themself in Twitter communities and seek to increase their followers by increasing the number of people following them.

### 4.1.2 Accounts Overlapping Perspective

Propagandists rely on the artificial amplification of Twitter interactions, including establishing several or overlapping accounts [39]. Figure 3 depicts the intersections between propagandist users in banking and sports topics. It illustrates that the propagandists' users are completely overlapped.

Only nine propagandist users participated in the computational propaganda sports campaign and did not participate in the banking campaign. That means the propagandist accounts used to manipulate public opinion on the banking topic were the same ones used to manipulate public opinion on the sports topics.

Regarding the non-propagandists' users, figure 3 shows that the users who post on both topics are only 91 accounts. So, this finding assures that to evaluate the account's reliability, it is important to understand the topicsthe author addresses and his stance [40].

### 4.1.3 Accounts Overlapping Perspective

The account creation date was not widely investigated in respect of propaganda detection. This research assumes that the age of the Twitter account is an indicator that may be considered when detecting propagandists' accounts [41].

The reason is that, based on previous studies, propaganda campaigns are carried out by an electronic army created for this purpose. Thus, the army will most likely be established in the same year the propaganda campaign is launched. The question may arise whether using the same army more than once is possible.

The sure answer is that we can use it technically, but Twitter makes a great effort to detect and close these accounts. Many of the accounts of former armies may have been eliminated. Figure 4 proves our assumption. It illustrates the account creation years of the propagandists' and non-propagandists' users.

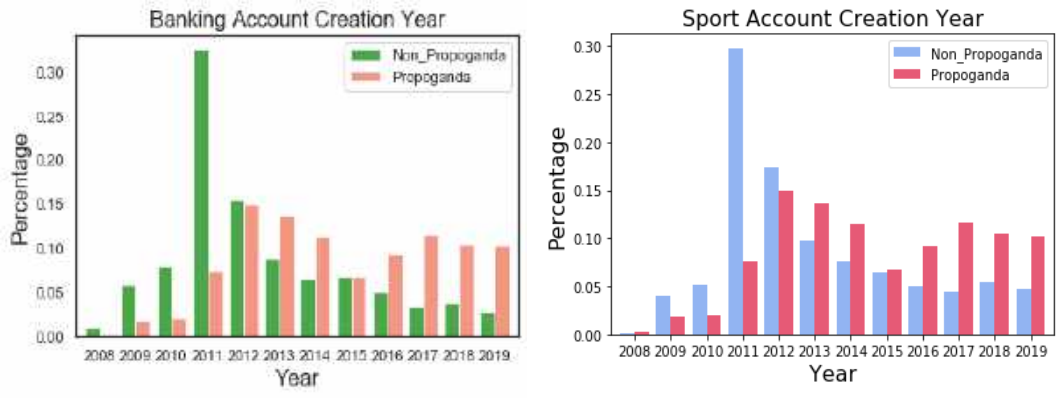



**Fig. 4.** The accounts creation years of propagandist and non-propagandist users

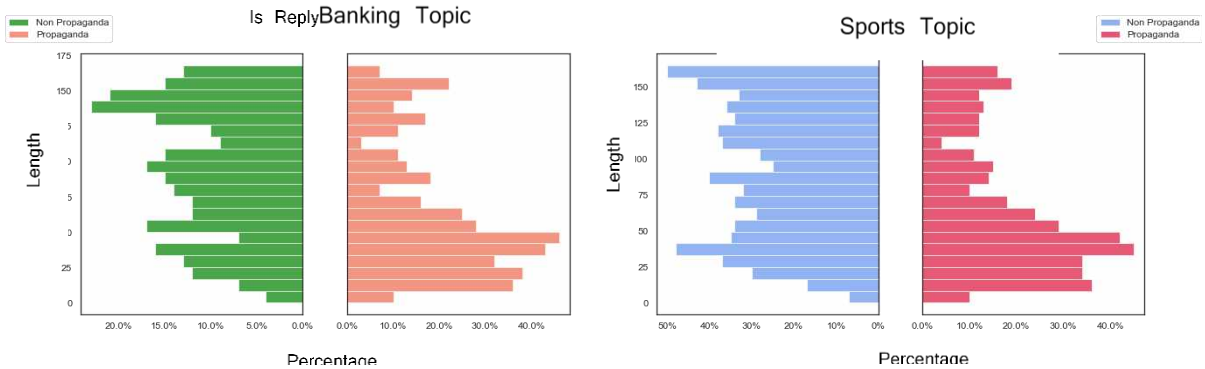



**Fig. 5.** Distributions of bio length of the propagandist and non-propagandist users in banking and sports topic

As shown in the figure, the propagandists' accounts are considered newer than non-propagandist accounts; about 40 percent were established in the last four years before the campaign.

### 4.1.4 Profile Description Perspective

The Twitter account profile description was analyzed as a feature often ignored in relevant works [42]. At this stage of the analysis, the content of the description in terms of the number of characters, the occurrence of hashtags, and the occurrence of URLs were considered. Figure 5 illustrate the differences in bio length between propagandist and non-propagandist users.

As shown in the figure, 10 per cent of the propagandist users do not provide any description in their profile for banking and sports topics. On the other hand, only about 3 per cent of banking non-propagandists and 4 per cent of sport non-propagandists do not describe themselves in their profiles.

In general, non-propagandists tend to describe themselves in long sentences more than propagandists

## 4.2 Tweet Level

To understand the pattern of computational propaganda, the EDA has been conducted at the tweet level. The analysis included several perspectives: first, it explored the tweets' originality by determining whether the propagandists' tweets were original or interactive. Second, it compared

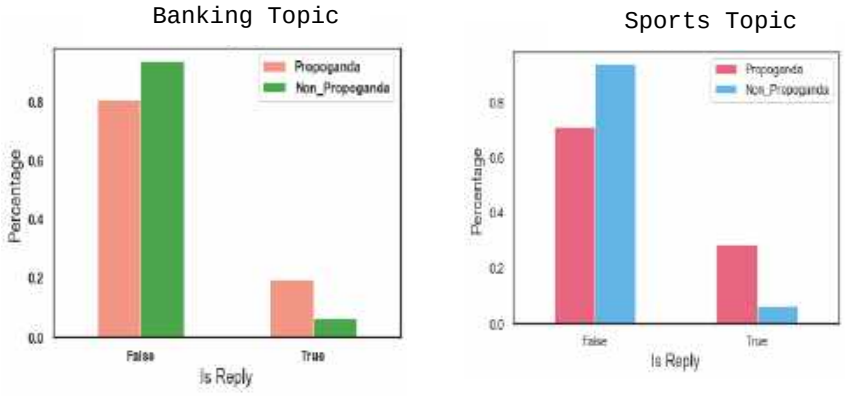



**Fig. 6.** Percentage of replies to tweets

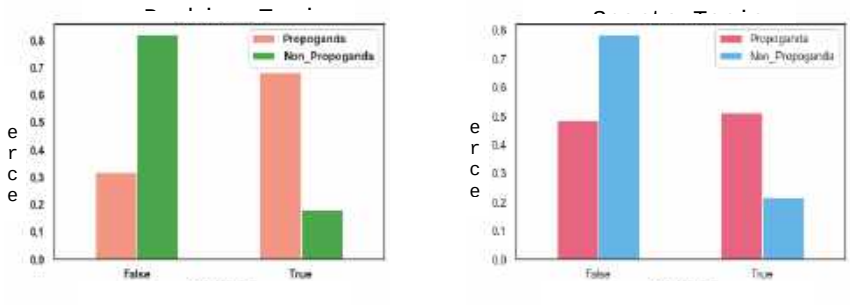



**Fig. 7.** Percentage of retweeted tweets

propagandists' tweets with non-propagandists' tweets regarding their ability to ignite interaction with the other users.

Third, it discovers their behaviour of embedding URLs and hashtags in tweets. Finally, it compares the daily publishing rates of propagandists and non-propagandists.

### 4.3 Tweet Originality Perspective

This analysis aims to explore the tweets' originality. Such analysis provides insights into whether to agree or refuse an existing approach. It assumes that postings from propagandists' accounts are almost identical since the supporters frequently pre-write the material [9, 10].

Figures 6 and 7 compare propagandist and non-propagandist patterns of originality in tweets on sports and banking topics. In general, we found that the pattern is similar in both topics. The propagandists' users tend to retweet the propagandists' tweets more than reply to them.

According to Figure 6, 20 percent of propagandists' banking tweets and 29 percent of propagandists sport tweets are replies, implying that some propagandist users try to engage in discussions by replying to each other. Based on the finding of Pacheco et al. [19], this behaviour increases the dynamicity of the propagandist's user clusters. As such, the community structure used for interaction in the network is highly affected by the discussed topic, altering the users' perceived affiliation.

However, we do not find many propagandists' tweets replying to other propagandists' tweets because the automated propagandist's accounts are just looking for a tale based on a narrative (or are expressly directed to a tweet or story). Moreover, they do not reply since they are unlikely to have NLP-Generation capabilities.

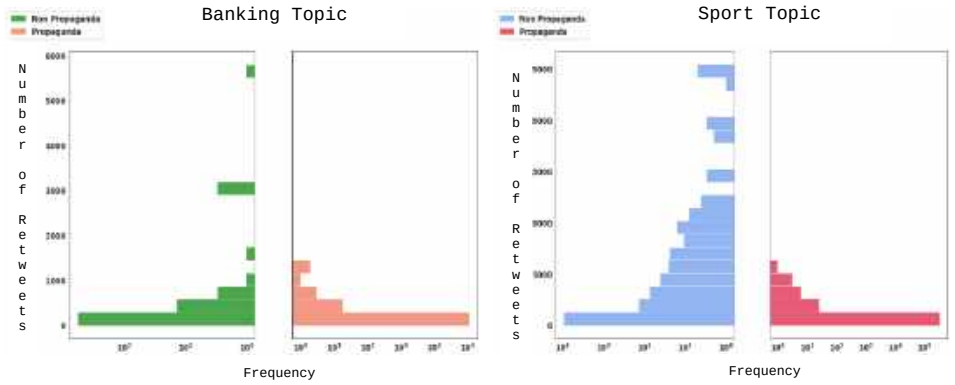



**Fig. 8.** Distributions of retweets interactions

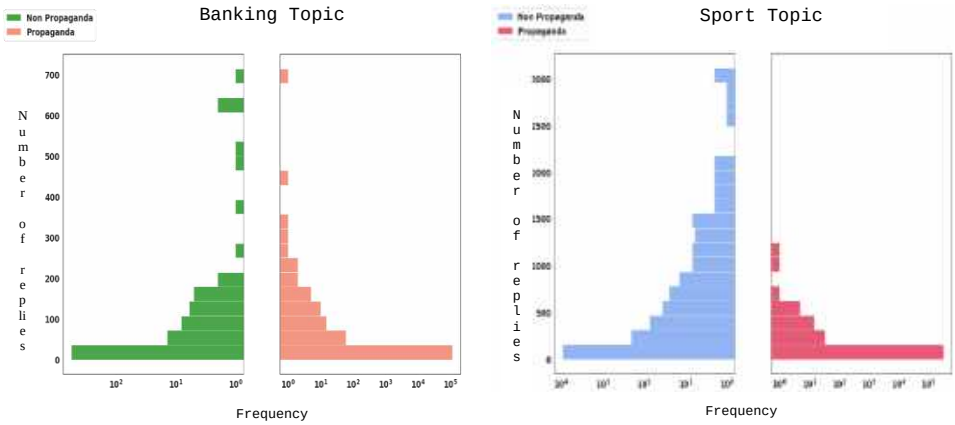



**Fig. 9.** Distributions of replies to interactions

Figure 7 shows that retweeting is a clear feature of propagandists' users: 68 percent of banking propagandists' tweets and 51 percent of sports propagandists' tweets are retweets.

This insight supports the finding of Pacheco et al. [19] that propagandists' tweets tend to duplicate and retweet content automatically to ensure that accounts work in a coordinated network to promote an account or story. Eventually, 81 percent and 78 percent of the non-propagandists' tweets on banking and sports topics are original, and they interact with each other in a normal pattern.

### 4.4 Engagement Rate Perspective

This aspect of the EDA attempts to compare the patterns of interactions ignited by both propagandists' and non-propagandists' tweets. It tracks how many retweets and replies each tweet receives in both topics. This will provide valuable insight as it determines the strength of a tweet's impact. As is customary, people click on links they believe will interest them.

They favour posts that they think are particularly noteworthy. However, they retweet anything that they believe would be of interest to their followers [5, 41]. Figure 8 shows the distribution of the gained retweets. The retweet amount of the sports non-propagandists' tweets is high, proving the finding of Tristan Handy [38], who found that 21 of 50 retweet accounts were sports-focused on social activity.

Furthermore, the figure shows that non-propagandists' tweets were continuously

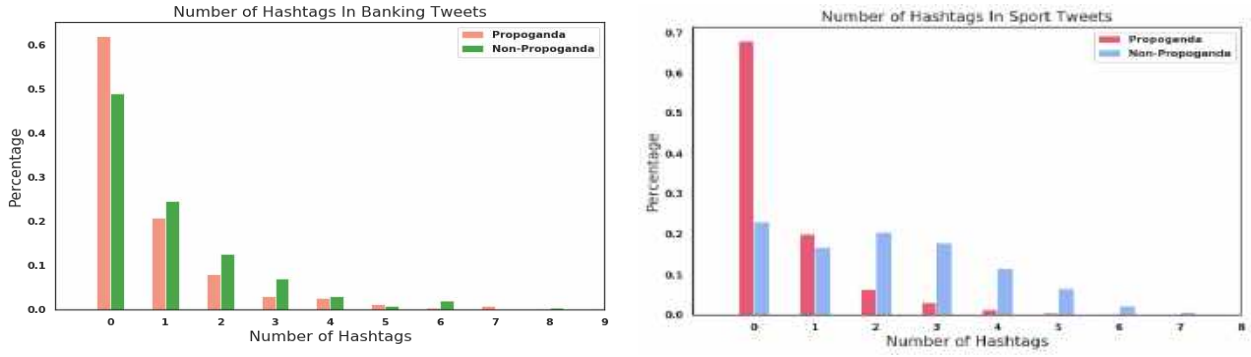



**Fig. 10.** Distributions of Hashtags embedded in tweets

retweeted on both topics, whereas propagandists' tweets were not frequently retweeted. The reason may be that the propagandists' tweets were originally retweets, as we mentioned above, and these retweets were not retweeted again. These notes reinforce the broker communication strategy discussed by Agarwal et al. [9]. Weber and Neumman [39] concluded there is a higher likelihood of mentioning or retweeting propaganda in a large community of propagandists internally.

Thus, we can assume that the propagandist army retweeted the tweets of their broker. As a result, a conceptual connection of users represents the topological organization, while the network clusters are highly polarized compared to clusters represented in the entire network [2]. In other words, the cluster of propagandists has the structure of a partisan community and a similar cluster of users with the same user identity.

Regarding participating in discussions, most of the replies on Twitter were related to sports topics. As shown in Figure 9, the non-propagandists' tweets ignited more replies on sports topics, although the number of propagandists' tweets is much higher than the propagandists' tweets in the dataset. Note that the replies do not have to be from non-propagandist users, as this is the goal of propaganda campaigns: to engage people in discussions to manipulate their opinions.

On the other hand, replies may be one of their strategies to influence the dynamics of the community structure [19], altering the users' perceived affiliation. So, we cannot consider the amount of replies as a distinguishing feature between propagandists' and non-propagandists' users.

### 4.4.1 Text Content Perspective

A hashtag is created on Twitter by adding a "#" to the beginning of an unbroken word or phrase. When a hashtag is included in a tweet, it links to all the other tweets that use it. Figure 10 illustrates the distribution of the hashtags used in propagandists' and non-propagandists' tweets on sports and banking topics.

The figure shows that non-propagandists users tend to embed hashtags in their tweets more than propagandist users. This can be attributed to the fact that including a hashtag in a tweet provides an explanation and helps users readily track subjects of interest.

As shown in Figure 10, 62 percent and 67 percent of the propagandists' tweets did not include hashtags in the banking and sports topics, respectively. On the other hand, 49 percent and 23 percent of the propagandists' tweets did not include hashtags in the banking and sports topics, respectively.The similarity of this behaviour between propagandists in both topics reflects the following behaviour.

However, these findings contradict Agarwal et al. [9],who found that propagandist posts are usually bracketed within hashtags. However, these findings prove that the propagandists change their strategies every time they deploy a new campaign [5]. Usually, a link is placed in the tweet when we want to direct users to a site outside of Twitter.

Figure 11 illustrates the distribution of URLs used in propagandists' and non-propogandists'

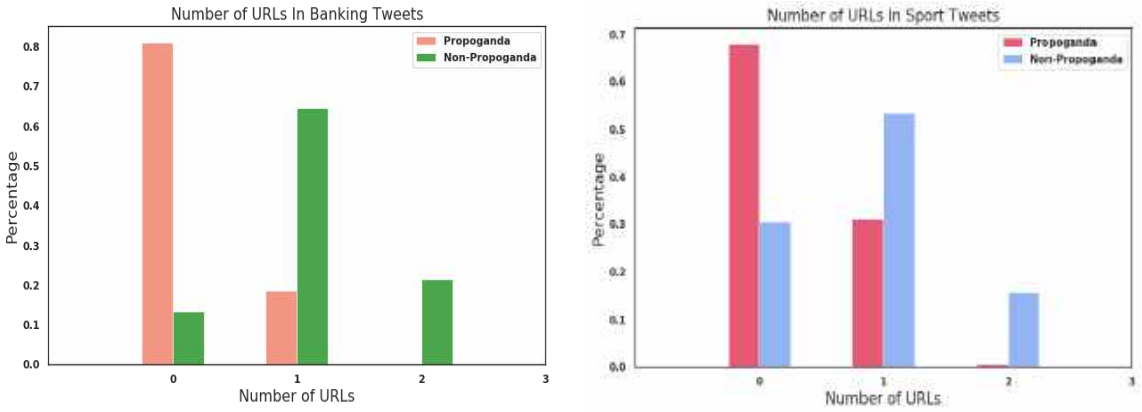



**Fig. 11.** Distributions of URLs embedded in tweets.4.2.4. Publishing time perspective

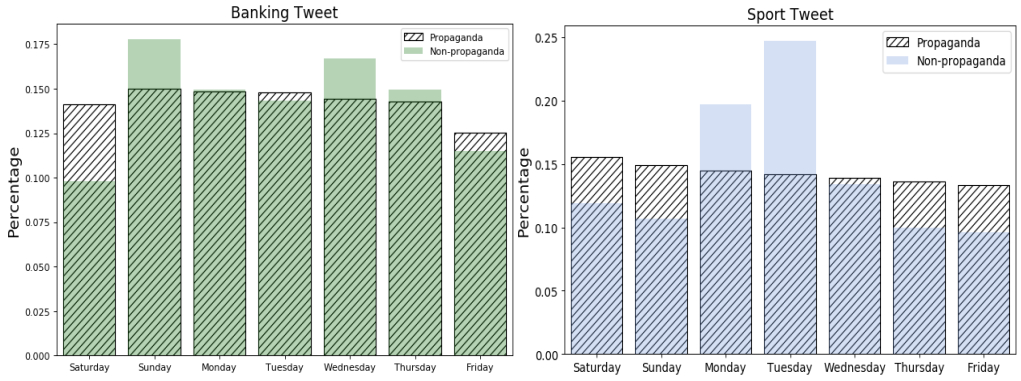



**Fig. 12.** Distributions of publishing days

tweets on sports and banking topics. It seems that non-propogandists' tweets include URLs more than propogandists' tweets.

Approximately 81 percent of propagandists' tweets on banking topics and 67 percent on sports topics did not include URLs. Almost all of the remaining propagandist tweets contain just one URL. There are very few tweets on the sports topic that include two URLs. On the contrary, URLs appear in 87 percent of non-propagandist tweets in banking topics and 69 percent in sports topics.

According to Agarwal et al. [9], the propagandists' tweets mostly contain links that lead users to the same article on a specific external website.

This finding contradicts ours, demonstrating no clear understanding of computational propaganda strategies because propagandist users change their strategy to manipulate public opinion at each campaign deployment [5].

### 4.4.2 Publishing Time Perspective

People log in to Twitter at various times based on their schedules and time zones. To reach a broad portion of the audience, it is necessary to determine what time of day the audience likes to view tweets so that they may be published properly. Choosing the right timing is certainly one of the strategies of the propagandists' users.

The time was analyzed for publishing days and periods to discover the propagandists' behaviours. Figure 12 investigates the publishing days. In both topics, propagandists' users do not differentiate between the days of the week; they post throughout the week almost equally as if they have a task to accomplish.

This behaviour is completely different from that of non-propagandists' users, as shown in the figure. Therefore, the number of tweets circulated on a suspicious topic throughout the day is an important indicator for detecting propaganda. The posting period was examined to obtain the most accurate results.

The posting timespan in the dataset has been modified to match the timing of the Kingdom of Saudi Arabia, which was targeted in this propaganda campaign. Figures 13 and 16 show the distribution of the posting timespan.

The result was very strange. The posting timespan pattern for banking andsports topics varied to match the posting timespan of non-propagandists' users interested in the same topic. We can conclude that the timing of propaganda campaigns is analyzed carefully to target the segment interested in the topic.

Therefore, the timespan is an important indicator for detecting propaganda while considering the topic.

#### 4.4.3 Publication Frequency Perspective

While excellent content is unquestionably the most crucial aspect of constantly increasing users' presence on social media, understanding how frequently to publish is critical to ensuring that users reach their target audience [43]. It is typically advised to publish no more than 1-2 times per day and no more than 3-5 times per day. Data reveals that engagement drops dramatically after the third tweet of the day [44]. Figure 14 depicts the distribution of daily posting times in both topics between propagandists and non-propagandists.

According to the figures, the number of daily tweets by non-propagandist users on both topics did not exceed 8 posts and 43 posts per day on the banking and sports topics, respectively [45].

Most of them publish only one tweet per day. As previously stated, Twitter users are typically active in sports topics due to their competitive nature. Therefore, we find that the number of daily tweets on sports topics exceeds those on banking topics.

Regarding propagandist users, daily posts reach 121 posts on the banking topic and 308 on the sports topic. Less than 50and 30 per cent publish one daily tweet on banking and sports topics, respectively.

That means propagandists rely on the artificial amplification of Twitter postings. These findings agree with those of Weber et al. [10] and Agarwal et al. [9] found that propagandist posts are usually made at high frequency, whereby many posts are made within a short period in a way that is not humanly possible.

## 5 Discussion, Result and Future Works

This research aims to reveal the characteristics of propagandist users and their posts on Twitter. It goes over the dataset collection and exploration process. The data-gathering procedure collects tweets from Twitter to create ground truth. Detecting propagandists on Twitter necessitates information on their accounts, tweets, and activity.

The propagandist dataset, which includes the propagandists' user-profiles and tweets, was requested from Twitter. However, to properly grasp their features and behaviours, they must be compared to reliable accounts' features. This entails determining Saudi official news ecosystem stakeholders' accounts.

Then, they gather their tweets on the same topics the propagandists' users discussed. An unsupervised technique was utilized to extract the primary topics discussed by propagandists. Eventually, two topics were selected to be investigated: sports and banking, since they represent the largest percentages of the dataset.

The related keywords were then extracted from these topics using the BERT approach. The final keywords were then refined concerning their keyness properties. Finally, the resulting keywords were used to crawl official users' tweets that discussed the same topics as propagandists' users in the same timeframe.

Exploratory Data Analysis (EDA) was conducted to analyze and discover propagandist users' main characteristics and tweets. It helps to discover data patterns, spot anomalies, and make assumptions. The analysis revealed that propagandists were primarily amplifying content beneficial to their clients, mostly through inauthentic engagement strategies such as retweeting and duplicating content.

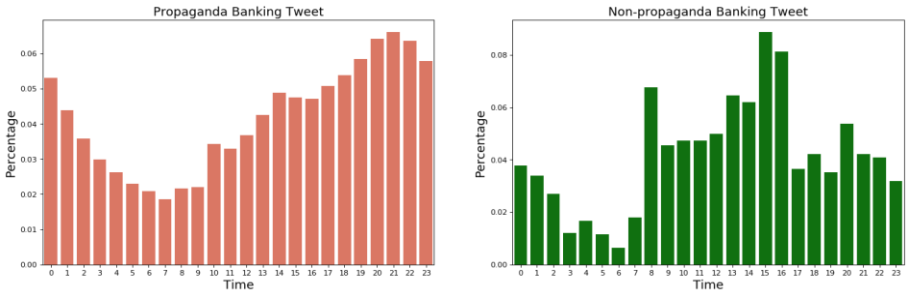



**Fig. 13.** Distributions of posting timespan in banking topic

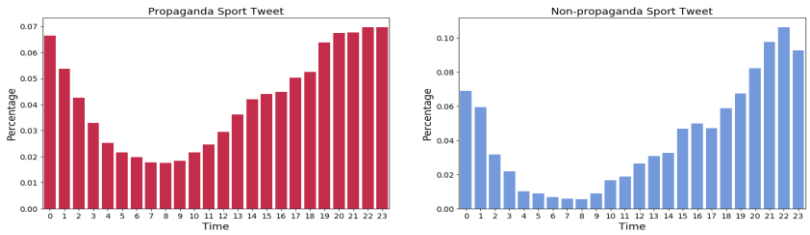



**Fig. 14.** Distributions of posting timespan in sport topic.

This insight supports Pacheco et al.'s [19] finding that propagandists' tweets tend to duplicate and retweet content automatically to ensure that accounts work in a coordinated network to promote an account or story. Since ancient times, repetition has been considered one of the techniques used to spread propaganda, but in the social media era, it is the most widely adopted technique [4].

What made matters worse was engaging many of the automated propagandist accounts to publish a mix of propagandist and non-propagandist content in enormous quantities. Usually, propagandists publish a mix of propaganda and non-propaganda tweets to hide their identities [4]. Using automation to tweet useful content does not violate Twitter's rules.

However, this behaviour was used carefully to conceal the larger platform manipulation perpetrated. This approach makes identifying propagandist tweets in the timelines of accounts that largely share automated, non-propagandist content more challenging.

The analysis was conducted on the user profile and tweet levels for a more in-depth investigation. In general, the analysis revealed some features of propagandist users discovered that are surprisingly different from those reported in other research.

For example, Agarwal et al. [9] found that propagandist posts are usually bracketed within hashtags, which contradicts the results of the analysis of this research. However, these findings prove that the propagandists change their writing style whenever they deploy a new campaign [5]. The analysis at the user level revealed important points. First, propagandists' users are greatly seeking to increase their popularity.

This point has not been investigated widely in previous literature [9]. The analysis revealed that propagandists implant themselves in Twitter communities by following a large segment of Twitter users because those following are expected to return the favour.

Thus, the number of followers can be increased by increasing the number of followers. In the social media era, the influence of a user can be associated with the number of their followers because their tweets reach a wide audience [9].

Second, the age of the account can be considered a feature that may help detect propagandists' users, although it was not widely investigated regarding propaganda detection.

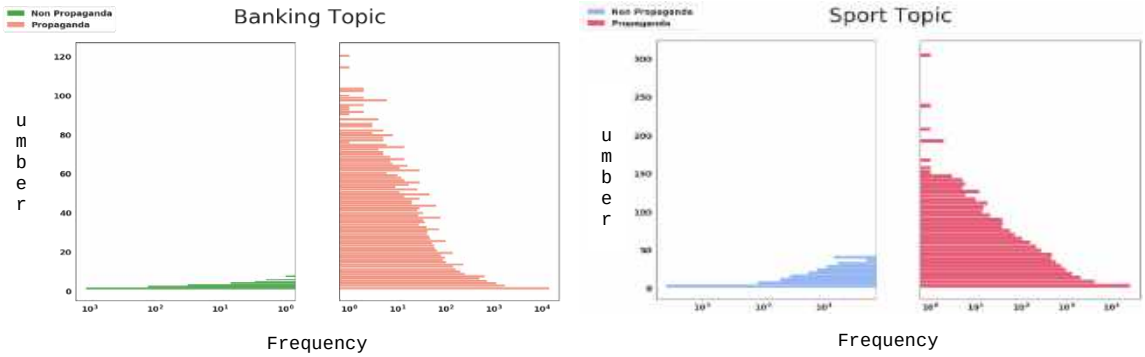



**Fig. 15.** Distributions of posting frequencies

Approximately 40 per cent were created four years before the campaign. Third, unlike propagandists, non-propagandists tend to describe themselves in their profiles with long sentences. Although the results show differences in profile length, we cannot consider him a propagandist user because he describes himself in a short sentence.

Many non-propagandists describe themselves in short sentences. To my knowledge, this feature has not been investigated widely [4]. Fourth, propagandists rely on the artificial amplification of Twitter interactions, including establishing several or overlapping accounts [39].

Regarding the analysis results related to the tweet level, we found thatmost propagandists' tweets are not original compared to non-propagandists' users. Propagandists' users tend to duplicate and retweet content automatically to ensure that accounts work in a coordinated network to promote a story, as stated by Agarwal et al. [9]. Second, propagandists rely on artificial amplification through widespread retweeting while posting little original content.

This result agrees with Guarino et al. [2], who found that the tweets of propagandists are retweets of their brokers. Third, propagandists tend to send a high volume of tweets in a short amount of time. Fourth, contradicting Agarwal et al. [9], the propagandists' users did not bracket their tweets with hashtags like the non-propagandists' users. They also did not embed URLs like the non-propagandist users.

Finally, the timespan is an important indicator for detecting propaganda if the campaign's topic has been considered.

Despite all the efforts, there is still no clear and definitive definition of what a malicious account looks like, which has created conflicting definitions. Unfortunately, research has shown that current technological weapons are used by malicious accounts just like their hunters, which increases their ability to escape detectors. Plus, a lack of real-ground truth allows for further investigation. We know that the data released by Twitter relates to propagandists.

However, at the level of their tweets, they are not classified; they mix propaganda and non-propaganda tweets to hide their identities. It would be very useful for the scientific community to classify propagandists' tweets to close the gap. The severity of this problem calls for special consideration from societies and research communities for a solution to be attained.

## 6 Conclusion

This research is motivated by the scarcity of research on Arab computational propaganda despite the significant increase in Arab social media users. Furthermore, the initial results motivate this research to enhance the previously cited efforts by discovering the features that can help detect Arab computer propaganda. It conducted a deep analysis to determine the propagandists' behavior and characteristics, regardless of their goals and writing style.

The results offer early evidence on social media regarding the propagandists' users and messages. Popularity, originality, topic diversity, publishing volume, and profile age metrics proved to be very informative features for detecting propaganda on Twitter. The oncoming research can combine these features with writing style features to develop a robust hybrid model.

## References

1. **Gil-de-Zúñiga, H., Jung, N., Valenzuela, S., (2012).** Social media use for news and individuals' social capital, civic engagement and political participation. Journal of Computer-Mediated Communication, Vol. 17, No. 3, pp. 319–336. DOI: 10.1111/j.1083-6101.2012.01574.x.
2. **Guarino, S., Trino, N., Celestini, A., Chessa, A., Riotta, G. (2020).** Characterizing networks of propaganda on Twitter: a case study. Applied Network Science, Vol. 5, No. 1, pp. 1–22. DOI: 10.1007/s41109-020-00286-y.
3. **Murphy, B. (2023**). Disinformation and democracy. Foreign Disinformation in America and the US Government's Ethical Obligations to Respond, pp. 25–35. DOI: 10.1007/978-3-031-29904-9_3.
4. **Howard, P., Lin, F., Tuzov, V. (2023).** Computational propaganda: Concepts, methods, and challenges. Communication and the Public, Vol. 8, No. 2, pp. 47–53. DOI: 10.1177/20570473231185996.
5. **Martino, G. D. S., Cresci, S., Barron-Cedeno, A., Yu, S., Di-Pietro, R., Nakov, P. (2020).** A survey on computational propaganda detection. IJCAI International Joint Conference on Artificial Intelligence, pp. 4826–32. DOI: 10.48550/arXiv.2007.08024b.
6. **Ferrara, E., Chang, H., Chen, E., Muric, G. Patel, J., (2020).** Characterizing social media manipulation in the 2020 US presidential election. First Monday, Vol. 25, No. 11. DOI: 10.5210/fm.v25i11.11431.
7. **Alvari, H., Sarkar, S., Shakarian, P. (2019).** Detection of violent extremists in social media. 2019 2nd international conference on data intelligence and security (ICDIS), IEEE, pp. 43–47. DOI: 10.1109/ICDIS.2019.00014.
8. **Chaudhari, D. D., Pawar, A. V. (2022**). A systematic comparison of machine learning and NLP techniques to unveil propaganda in social media. Journal of Information Technology Research (JITR), Vol. 15, No. 1, pp. 1–14. DOI: 10.4018/JITR.299384.
9. **Agarwal, N., Al-khateeb, S., Galeano, R., Goolsby, R. (2017).** A systematic comparison of machine learning and NLP techniques to unveil propaganda in social media. Defence Strategic Communications, Vol. 2, No. 2, pp. 87–112. DOI: 10.30966/2018.riga.2.4.
10. **Weber, D., Neumann, F. (2020).** Who's in the gang' revealing coordinating communities in social media. Proceedings of the 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 89–93. DOI: 10.1109/ASONAM49781.2020.9381418.
11. **Khanday, A. M., Khan, Q. R., Rabani, S. T. (2021).** Identifying propaganda from online social networks during COVID-19 using machine learning techniques. International Journal of Information Technology, Vol. 13, No. 1, pp. 115–22. DOI: 10.1007/s41870-020-00550-5.
12. **Da-San-Martino, G., Yu, S., Barrón-Cedeño, A., Petrov, R., Nakov, P. (2019).** Fine-grained analysis of propaganda in news article. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 5635–5645. DOI: 10.18653/v1/D19-1565v.
13. **Kumar, S., Cheng, J., Leskovec, J., Subrahmanian, V. S. (2017).** An army of me: sockpuppets in online discussion communities. Proceedings of the 26th International Conference on World Wide Web, pp. 857–866. DOI: 10.1145/3038912.3052677.
14. **Horne, B. D., Adalı, S. (2017).** This just in: fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. Proceedings of the International AAAI Conference on Web and

Social Media DOI: 10.1609/icwsm.v11 i1.14976.

15. **Abdullah, M., Altiti, O., Obiedat, R. (2022).** Detecting propaganda techniques in english news articles using pre-trained transformers. 13th International Conference on Information and Communication Systems, pp. 301–308. DOI: 10.1109/ICICS55353.2022.9811117.
16. **Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., Choi, Y., Allen, P. G. (2017).** Truth of varying shades: Analyzing language in fake news and political Fact-Checking. 2017 conference on empirical methods in natural language processing. pp. 2931–2937. DOI: 10.18653/v1/D17-1317.
17. **Horne, B. D., Adali, S. (2017)**. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. Vol. 11, No. 1, pp. 759–766. DOI: 10.18653/v1/D17-1317.
18. **Nerino, V. (2021).** Tricked into supporting: A study on computational propaganda persuasion strategies. Italian Sociological Review, Vol. 11, No. 3. DOI: 10.13136/isr.v11 i4S.438.
19. **Pacheco, D., Flammini, A., Menczer, F. (2020).** Unveiling coordinated groups behind white helmets disinformation. The Web Conference 2020, Companion of the World Wide Web Conference, pp. 611–616. DOI: 10.1145/3366424.3385775.
20. **Darwish, K., Alexandrov, D., Nakov, P., Mejova, Y. (2017).** Seminar users in the arabic twitter sphere. Lecture Notes in Computer Science [Internet], Springer, pp. 91–108. DOI: 10.1007/978-3-319-67217-5_7.
21. **Howard, P., Lin, F., Tuzov, V., (2023).** Computational propaganda: Concepts, methods, and challenges. Communication and the Public, Vol. 8, No. 2, pp. 47–53. DOI: 10.1007/978-3-319-67217-5_7.
22. **Wiard, V. (2019).** News ecology and news ecosystems. Oxford Research Encyclopedia of Communication, DOI: 10.1093/acrefore/978 0190228613.013.847.
23. **Barbas, Á., Trere, E. (2023).** The rise of a new media ecosystem: exploring 15M's educommunicative legacy for radical democracy. Social Movement Studies, Vol. 22, No. 3, pp. 381–401. DOI: 10.1080/14742837. 2022.2070738.
24. **Jarwar, M. A., Abbasi, R. A., Mushtaq, M., Maqbool, O., Aljohani, N. R., Daud, A., Chong, I. (2017).** CommuniMents: A framework for detecting community based sentiments for events. International Journal on Semantic Web and Information Systems, Vol. 13, No. 2, pp. 87–108. DOI: 10.4018/IJSWIS. 2017040106.
25. **Firoozeh, N., Nazarenko, A., Alizon, F., Daille, B. (2020).** Keyword extraction: Issues and methods. Natural Language Engineering [Internet], Vol. 26, No. 3, pp. 259–291. DOI: 10.1017/S1351324919000457.
26. **Giarelis, N., Kanakaris, N., Karacapilidis, N. (2021).** A Comparative assessment of state-of-the-art methods for multilingual unsupervised keyphrase extraction. IFIP Advances in Information and Communication Technology, Springer International Publishing, pp. 635–645. DOI: 10.1017/S1351324919000457.
27. **Vargas-Calderón, V., Dominguez, M. S., Parra-A, N., Vinck-Posada, H., Camargo, J. E. (2020).** Using machine learning techniques for discovering latent topics in twitter colombian news. Communications in Computer and Information Science, pp. 132–141. DOI: 10.1080/10810730.2018.1423648.
28. **Mannor, S., Jin, X., Han, J., Jin, X. (2011).** K-means clustering. Encyclopedia of Machine Learning, pp. 563–564. DOI: 10.1007/978-0-387-30164-8_425.
29. **Lulu, L., Elnagar, A. (2018).** Automatic arabic dialect classification using deep learning models. Procedia Computer Science, Vol. 142, pp. 262–269. DOI: 10.1016/j.procs.2018. 10.489.
30. **Bojanowski, P., Grave, E., Joulin, A., Mikolov, T. (2017).** Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, Vol. 5, pp. 135–146. DOI: 10.1162/tacl_a_00051.
31. **Newling, J., Fleuret, F. (2016**). Nested mini-batch K-Means. Advances in Neural Information Processing Systems, pp. 1360–1368. DOI: 10.5555/3157096.

**32. Wang, S., Li, H. (2020).** Adaptive K-valued K-means clustering algorithm. Proceedings 2020 5th International Conference on Mechanical, Control and Computer Engineering, pp. 1442–1445. DOI: 10.1109/ICMCCE51767.2020.00316.

**33. Pazos-Rangel, R. A., Florencia-Juarez, R., Paredes-Valverde, M. A., Rivera, G. (2020).** Handbook of research on natural language processing and smart service systems. IGI Global. DOI: 10.4018/978-1-7998-4730-4.

**34. Devlin, J. (2018).** Bert: Pre-training of deep bidirectional transformers for language understanding. DOI: 10.18653/v1/n19-1423.

**35. Pontius, R. G., Millones, M. (2011**). Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment. International Journal of Remote Sensing, Vol. 32, No. 15, pp. 4407–4429. DOI: 10.1080/01431161.2011.552923.

**36. Rousseau, F., Vazirgiannis, M. (2015).** Main core retention on graph-of-words for single-document keyword extraction. Advances in Information Retrieval, Springer, pp. 382–393. DOI: 10.1007/978-3-319-16354-3_42.

**37. Al-Zoubi, A. M., Alqatawna, J., Faris, H. (2017).** Spam profile detection in social networks based on public features. 2017 8th International Conference on Information and Communication Systems, pp. 130–135. DOI: 10.1109/IACS.2017.7921959.

**38. Pancer, E., Poole, M., (2016).** The popularity and virality of political social media: Hashtags, mentions, and links predict likes and retweets of 2016 US presidential nominees' tweets. Social Influence, Vol. 11, No. 4, pp. 259–270. DOI: 10.1080/15534510.2016.1265582.

**39. Weber, D., Neumann, F. (2021).** Amplifying influence through coordinated behavior in social networks. Social Network Analysis and Mining, Vol. 11, No. 1, pp. 1–42. DOI: 10.1007/s13278-021-00815-2.

**40. Hardalov, M., Arora, A., Nakov, P., Augenstein, I. (2021).** A Survey on stance detection for mis and disinformation identification. Findings of the Association for Computational Linguistics, pp. 1259–1277. DOI: 10.48550/arXiv.2103.00242.

**41. Ilias, L., Roussaki, I. (2021).** Detecting malicious activity in Twitter using deep learning techniques. Applied Soft Computing, Vol. 107, p. 107360. DOI:10.1016/J.ASOC.2021.107360.

**42. Hardalov, M., Arora, A., Nakov, P., Augenstein, I. (2021).** A survey on stance detection for mis- and disinformation identification. Findings of the Association for Computational Linguistics: NAACL 2022 Findings, pp. 1259–1277. DOI: 10.48550/arXiv.2103.00242.

**43. Hayawi, K., Mathew, S., Venugopal, N., Masud, M. M., Ho, P. H. (2022).** DeeProBot: a hybrid deep neural network model for social bot detection based on user profile data. Social Network Analysis and Mining, Vol. 12, No. 1, pp. 1–19. DOI: 10.1007/s13278-022-00869-w.

**44. Farivar, S., Wang, F., Turel, O. (2022).** Followers' problematic engagement with influencers on social media: An attachment theory perspective. Computers in Human Behavior, Vol. 133, p. 107288. DOI: 10.1016/j.chb.2022.107288.

**45. Valencia-Segura, K. M., Escalante, H. J., Villasenor-Pineda, L. (2023).** Automatic depression detection in social networks using multiple user characterizations. Computación y Sistemas, Vol. 27, No. 1, pp. 283–294. DOI: 10.13053/cys-27-1-4540.

*Article received on 25/02/2024; accepted on 15/05/2024.*
**Corresponding author is Bodor Moheel Almotairy.*

ISSN 2007-9737

# An Extension of the ELECTRE III Method based on the 2-tuple Linguistic Representation Model for Dealing with Heterogeneous Information

Juan Carlos Leyva-López[1,*], Jesús Jaime Solano-Noriega[2], Jorge Anselmo Rodríguez-Castro[1], Pedro J. Sánchez[3]

[1] Universidad Autónoma de Occidente, Culiacán, México

[2] Universidad Autónoma de Occidente, Los Mochis, México

[3] Universidad de Jaén, Spain

{juan.leyva, jaime.solano, jorge.rodriguez}@uadeo.mx, pedroj@ujaen.es

**Abstract.** Multicriteria decision analysis (MCDA) is a problem-solving approach that helps to tackle complex decision-making problems. It involves analyzing a set of alternatives that are assessed based on a set of decision criteria by one or more decision-makers. These decision-makers use both subjective and objective judgments, which can be qualitative and/or quantitative. The goal of MCDA is to arrive at a decision that is fair, effective, and considers all relevant factors. Some MCDA methods lack mechanisms to consistently process heterogeneous information provided by the decision-maker and reduce it simplistically to numerical values to assess subjective criteria and thus obtain numerical results with low interpretability. This paper presents an extension of the ELECTRE III method that considers heterogeneous information provided by the decision-maker as input data in the decision criteria. The new proposal is based on the 2-tuple linguistic representation model, which allows for a flexible assessment structure in which the decision-maker can provide their preferences by applying diverse levels of information according to the nature and uncertainty of the decision criteria. It includes a new distance measure based on linguistic transformations appropriate for the MCDA outranking approach. Finally, the viability and pertinence of the proposed method are shown in a case to evaluate the environmental impact that can occur between the interactions of some industrial activities in a petrol station.

**Keywords.** Computing with words, heterogeneous information, linguistic preferences, multicriteria decision analysis, ELECTRE III.

## 1 Introduction

Multicriteria Decision Analysis (MCDA) provides a methodological framework for managing complex decision-making problems with multiple criteria in conflict. The purpose and scope of MCDA are to support decision-makers (DMs) while addressing complex decision-making problems.

The MCDA outranking approach involves ranking a set of alternatives in decreasing order of preferences. This is a multicriteria ranking problem, where there may be ties and incomparability among the alternatives [1, 2]. The ranking means a recommendation for the DM generated by the solution method.

The MCDA outranking methods combine the aggregation and exploitation phases. In the preference aggregation phase, the DM's preference aggregation model is achieved, represented by an outranking relation, which usually does not present attractive mathematical

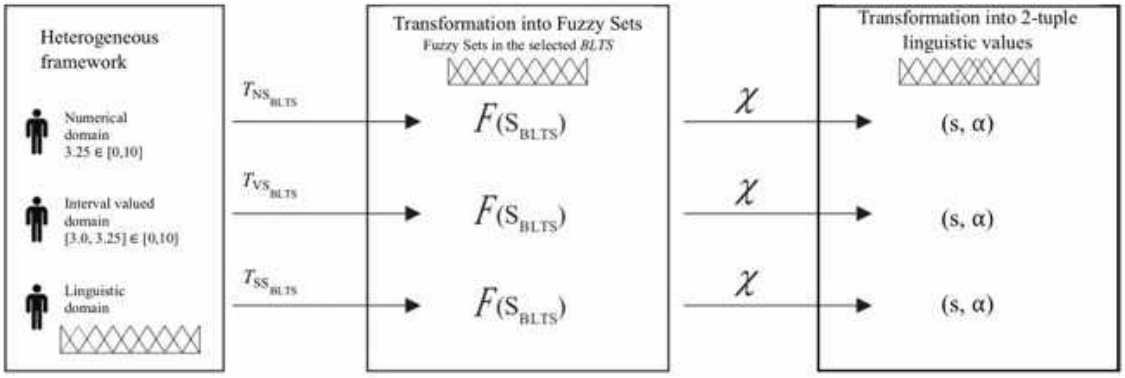



**Fig. 1.** Fusion approach dealing with heterogeneous information using the 2-tuple fuzzy linguistic model

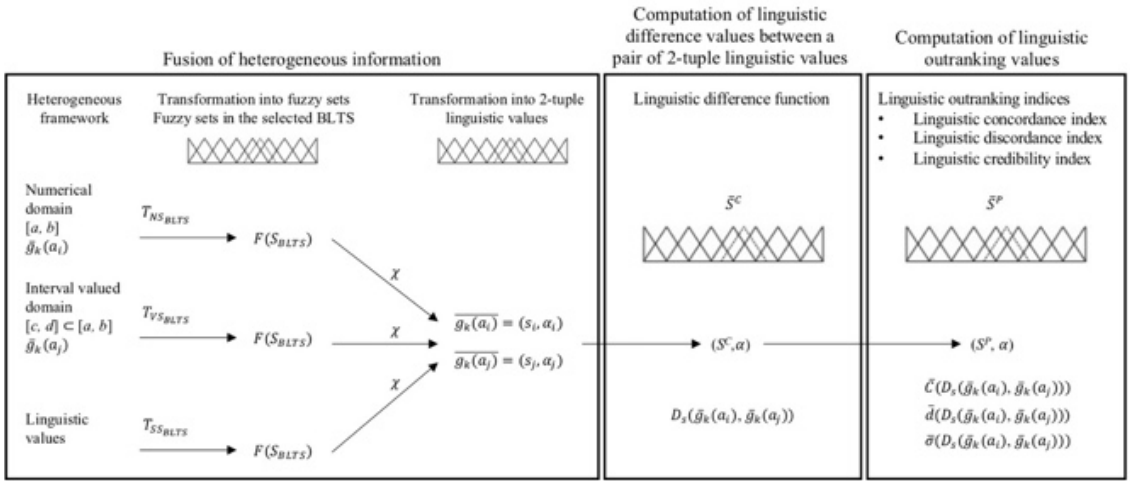



**Fig. 2.** Process for modeling linguistic outranking indices

properties such as transitivity and completeness [3].

In the exploitation phase, a partial preorder is deduced from the outranking relation, reflecting irreducible incomparability and indifference between the alternatives [4].

ELECTRE III [5] is a representative method of MCDA that constructs and exploits a fuzzy outranking relation.

The ELECTRE III method has been widely used to solve many real-world problems that can be formulated as multicriteria ranking problems. However, like any multicriteria method, it has weaknesses and limitations when using it for specific instances of the ranking problem. For example, ELECTRE III is inadequate regarding:

- Dealing with heterogeneous information: DMs must use numerical scales in ELECTRE III, which is inflexible as criteria can have varied descriptions and be assessed in different expression domains.
- Dealing with uncertainty: ELECTRE III cannot correctly handle the uncertainties and vagueness of subjective judgments.

This paper proposes an extension of the ELECTRE III method to reduce limitations in information management, incorporating a flexible heterogeneous evaluation structure in the decision criteria to analyze elements of uncertainty and vagueness that occur in many instances of the multicriteria ranking problem, which is more in line

**Table 1.** Criteria set description

| | Name | Description | Expression Domain |
|---|---|---|---|
| $g_1$ | Intensity | How the action impacts the factor | Linguistic: L |
| $g_2$ | Extension | The range within which the action affects the site | Linguistic: L |
| $g_3$ | Moment | The duration from the action's onset to the time the factor begins to be affected | Numerical: N |
| $g_4$ | Persistence | The estimated duration of the action's effect | Valor Interval-valued: I |
| $g_5$ | Reversibility | The potential for the factor to be naturally restored to its original state after being affected | Numerical: N |
| $g_6$ | Synergy | The strengthening of straightforward impacts | Linguistic: L |
| $g_7$ | Accumulation | The gradual escalation in the expression of the effect | Linguistic: L |
| $g_8$ | Effect | How the action's impact on an environmental factor becomes apparent | Linguistic: L |
| $g_9$ | Recoverability | The potential for the factor to be restored with the help of human action | Linguistic: L |
| $g_{10}$ | Periodicity | The consistency in which the impact on the environmental factor is observed | Linguistic: L |

**Table 2.** Assessment for each criterion on each *EI* using a heterogeneous framework

| | $g_1$ | $g_2$ | $g_3$ | $g_4$ | $g_5$ | $g_6$ | $g_7$ | $g_8$ | $g_9$ | $g_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $ei_1$ | L | L | 0,00 | [0,0.2] | 1,00 | L | L | L | H | VL |
| $ei_2$ | H | VL | 0,00 | [0,0.2] | 1,00 | M | M | M | VH | VL |
| $ei_3$ | H | L | 0,00 | [0.4,0.6] | 10,00 | M | L | VH | L | M |
| $ei_4$ | M | VL | 0,10 | [0,0.2] | 2,00 | L | VL | VL | M | H |
| $ei_5$ | H | L | 0,10 | [0,0.05] | 10,00 | M | M | VH | L | M |
| $ei_6$ | VH | L | 0,00 | [0.8,1.0] | 10,00 | M | M | VH | L | H |
| $ei_7$ | VL | L | 1,50 | [0.8,1.0] | 10,00 | VL | L | VL | L | VH |

with the quantitative and qualitative essence of the decision criteria and with the experience of the DM.

The fusion linguistic approach converts heterogeneous information into a linguistic one [4, 5, 6]. The fusion approach for an MCDA method makes the computations possible and generates interpretable results [2]; however, it implies the need for utilizing Computing with Words (CW) procedures [7, 8].

Therefore, we employ the fuzzy linguistic approach based on the 2-tuple linguistic representation model [9, 10] and a linguistic-based distance measure [11, 12] to construct a fuzzy outranking relation in such a way that the DM can make his evaluations in different sets of linguistic terms according to his knowledge of the decision problem [6].

We consider a modified distillation procedure to derive a partial preorder of the alternatives for the exploitation phase of the fuzzy outranking relation based on 2-tuple linguistic representation modeling. This approach's main advantage is tackling the uncertainty of criteria performances and DMs' knowledge without losing information.

Thus, this new method will be helpful in multicriteria ranking problems whose DMs express their value judgments through heterogeneous values. In these kinds of issues, it is common for

**Table 3**. Fused information supplied by the DM

| | $g_1$ | $g_2$ | $g_3$ | $g_4$ | $g_5$ | $g_6$ | $g_7$ | $g_8$ | $g_9$ | $g_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $ei_1$ | (L,0) | (L,0) | (VH,0) | (VL,0.44) | (VL,0.4) | (L,0) | (L,0) | (L,0) | (H,0) | (VL,0) |
| $ei_2$ | (H,0) | (VL,0) | (VH,0) | (VL,0.44) | (VL,0.4) | (M,0) | (M,0) | (M,0) | (VH,0) | (VL,0) |
| $ei_3$ | (H,0) | (L,0) | (VH,0) | (M,0) | (VH,0) | (M,0) | (L,0) | (VH,0) | (L,0) | (M,0) |
| $ei_4$ | (M,0) | (VL,0) | (VH,-0.04) | (VL,0.44) | (L,-0.2) | (L,0) | (VL,0) | (VL,0) | (M,0) | (H,0) |
| $ei_5$ | (H,0) | (L,0) | (VH,-0.04) | (VL,0.17) | (VH,0) | (M,0) | (M,0) | (VH,0) | (L,0) | (M,0) |
| $ei_6$ | (VH,0) | (L,0) | (VH,0) | (VH,-0.44) | (VH,0) | (M,0) | (M,0) | (VH,0) | (L,0) | (H,0) |
| $ei_7$ | (VL,0) | (L,0) | (H,0.4) | (VH,-0.44) | (VH,0) | (VL,0) | (L,0) | (VL,0) | (L,0) | (VH,0) |

DMs to have different levels of knowledge and domains of the criteria.The organization for the rest of the document is presented as follows: Section 2 shows background about ELECTRE III and linguistic and heterogeneous information. Then, section 3 gives the linguistic ELECTRE III, and section 4 provides an illustrative example. Finally, section 5 points out the conclusions.

## 2 Background

This section briefly describes the ELECTRE III method and the use of linguistic and heterogeneous information in MCDA.

### 2.1 The ELECTRE III Method

The ELECTRE III method is a decision support technique under the category of outranking methods designed to solve problems involving multiple criteria [13, 14]. It is a multicriteria ranking method that is relatively simple in conception and application compared to other MCDA methods. It can be applied to situations where a finite set of alternatives must be prioritized, considering multiple, often contradictory, criteria (e.g. [15, 16]).

ELECTRE III needs a decision matrix with criteria evaluations for each alternative and preference information in the form of weights and thresholds. The model accounts for uncertainty in the assessments when defining the thresholds.

This method consists of two distinct stages. Initially, (*i*) it aggregates the input data to create a fuzzy outranking relation on the pairs of alternatives.

Subsequently, (*ii*) it exploits the fuzzy outranking relation to generate a partial preorder of alternatives [14]. Let us have the following notations:

$A = \{a_1, a_2, \dots, a_m\}$ is defined as the set of potential decision alternatives.

$G = \{g_1, g_2, \dots, g_n\}$ is defined as a coherent family of criteria.

$W = \{w_1, w_2, \dots, w_n\}$ is defined as the set of weights that reflects the DM's preferences over the set $G$. Let us assume that $\sum_{i=1}^{n} w_i = 1$.

$g_j(a_i)$ is the evaluation of the criterion $g_j$ for alternative $a_i$.

The ELECTRE III method uses the fundamental principle of threshold values; an indifferent $q$ threshold is determined as:

$$a_i P a_j \left(a_i \; is \; prefered \; to \; a_j\right) \leftrightarrow g(a_i) > g\left(a_j\right) + q.$$

And

$$a_i I a_j \left(a_i \; is \; indifferent \; to \; a_j\right) \leftrightarrow \left|g(a_i) - g\left(a_j\right)\right| \leq q.$$

The inclusion of the indifference threshold addresses the consideration of a DM's sentiment toward practical comparisons of the alternatives. However, a marker still exists when a DM's preference transitions from a state of indifference to a state of strict preference. Regarding conceptual understanding, it is beneficial to introduce a buffer region between indifference and strict preference.

**Table 4.** Indifference $q$ preference $p$ and veto $v$ values

| Criterion | $\left(\left(I^C_{t_{q_k}}, \alpha_{q_k}\right) q\right)$ | $\left(\left(I^C_{t_{p_k}}, \alpha_{p_k}\right) p\right)$ | $\left(\left(I^C_{t_{v_k}}, \alpha_{v_k}\right) v\right)$ |
|---|---|---|---|
| $g_1$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | $(I^C_7, 0)v$ |
| $g_2$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | $(I^C_7, 0)v$ |
| $g_3$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_4$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_5$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_6$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_7$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_8$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_9$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |
| $g_{10}$ | $(I^C_5, 0)q$ | $(I^C_6, 0)p$ | |

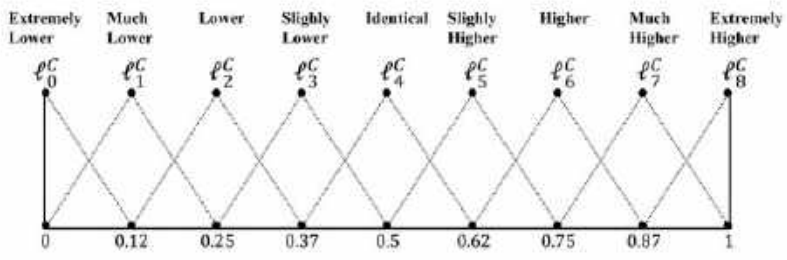



**Fig. 3.** Linguistic comparison scale $S^C$

This intermediary region represents a state where the DM hesitates over indifference and preference, known as a weak preference. Like the preference relations indifference ($I$) and strict preference ($P$), this zone of hesitation is modeled by introducing a preference threshold, denoted as $p$. Therefore, we adopt an indifference-preference model, incorporating a binary relation $Q$ for weak preference measurement:

$$\begin{aligned} &a_i P a_j \left(a_i \textit{ is strictly preferred to } a_j\right) \leftrightarrow g(a_i) - g(a_j) > p \\ &a_i Q a_j \left(a_i \textit{ is weakly preferred to } a_j\right) \leftrightarrow q < g(a_i) - g(a_j) \\ &\qquad \leq p \\ &a_i I a_j \left(a_i \textit{ is indifferent to } a_j; \textit{and } a_j \textit{ to } a_i\right) \\ &\qquad \leftrightarrow |g(a_i) - g(a_j)| \leq q \end{aligned} \tag{1}$$

The selection of thresholds significantly impacts the determination of specific binary relations. In [14, 17], detailed information is provided on how to compute thresholds in ELECTRE III, including their nature, meaning, and form.

We must acknowledge that we have solely examined the basic scenario where the thresholds $q$ and $p$ are constant values rather than functions dependent on the criteria's values. The ELECTRE method can be presented in a more straightforward way by using constant thresholds. However, employing variable thresholds in situations where criteria with higher values could result in more substantial indifference and preference thresholds might be advantageous.

ELECTRE III uses these thresholds in the aggregation procedure to create the outranking relation $O$. Based on the DM' preference model, we can justify that "$a_i$ is at least as good as $a_j$ "denoted as $a_i O a_j$. Subsequently, each pair of alternatives is evaluated to verify the validity of $a_i O a_j$ from which one of the following states can happen:

i) $a_i O a_j$ and $\neg(a_j O a_i)$; ii) $\neg(a_i O a_j)$ and $a_j O a_i$; iii) $a_i O a_j$ and $a_j O a_i$; and iv) $\neg(a_i O a_j)$ and $\neg(a_j O a_i)$. States iii and iv agree with the indifference and incomparability preference relations denoted as $I$ and $R$ *respectively*.

There are two principles that ELECTRE III incorporates to validate $a_i O a_j$, the concordance and the non-discordance principles. The former holds that most criteria, considering their

respective significance, support the assertion $a_i O a_j$; meanwhile, the second principle holds that a minority of the criteria are against $a_i O a_j$. These two principles are executed as follows: suppose criteria are to maximize; first, we examine the outranking relation established for each criterion where $a_i O_k a_j$ denotes "$a_i$ is at least as good as $a_j$" on criterion $k: k = 1,2, \dots, n$.

Let $q$ and $p$ be an indifference and preference thresholds, in this scenario, criterion $k$ is in concordance with $a_i O a_j$ iif $g_k(a_i) \geq g_k(a_j) - q_k$ (e.g., $a_i O_k a_j$). Conversely, criterion $k$ is in discordance with $a_i O_k a_j$ iif $g_k(a_j) \geq g_k(a_i) + p_k$ (e.g. $a_j P_k a_i$). Hence, using the concordance and discordance rules, the claim $a_i O_k a_j$ can be assessed.

The initial stage involves creating a concordance assessment, represented by the concordance index $C(a_i, a_j)$ for each $(a_i, a_j) \in A \times A$. Suppose that $w_k$ represents the weight for the $k - th$ criterion, the concordance index (Eq. 2) can be expressed as follows:

$$C(a_i, a_j) = \frac{1}{W} \sum_{k=1}^{n} w_k c_k(a_i, a_j), \tag{2}$$

where:

$$W = \sum_{k=1}^{n} w_k. \tag{3}$$

And:

$$c_k(a_i, a_j) = \begin{cases} 0, if\ g_k(a_i) + p_k \leq g_k(a_j), \\ \dfrac{p_k + g_k(a_i) - g_k(a_j)}{p_k - q_k}, \\ if\ g_k(a_j) - p_k \leq g_k(a_i) < g_k(a_j) - q_k, \\ 1, if\ g_k(a_i) + q_k \geq g_k(a_j), \end{cases} \tag{4}$$

where k = 1,2,…,n.

The second stage involves creating the discordance index, which integrates the veto $v$ threshold. With this threshold, it is possible to reject completely $a_i O a_j$ if, for any given criterion veto threshold, $v_k$, $g_k(a_j) > g_k(a_i) + v_k$. The calculation of the discordance index $d_k(a_i, a_j)$ (Eq. 5) for each criterion $k$ is performed as follows:

$$d_k(a_i, a_j) = \begin{cases} 0, if\ g_k(a_i) + p_k \geq g_k(a_j), \\ \dfrac{g_k(a_j) - g_k(a_j) - p_k}{v_k - p_k}, \\ if \\ g_k(a_j) - v_k < g_k(a_i) < g_k(a_j) - p_k \\ 1, if\ g_k(a_i) + v_k \leq g_k(a_j), \end{cases} \tag{5}$$

where k=1,2 ,..., n.

Finally, both measures, concordance, and discordance, must be fused to make a metric that reflects the power of the affirmation $a_i S a_j$. This metric is known as the credibility index $\sigma(a_i, a_j) (0 \leq \sigma(a_i, a_j) \leq 1)$ and is stated in (Eq. 6) as:

$$\sigma(a_i, a_j) = \begin{cases} C(a_i, a_j), if\ K(a_i, a_j) = \emptyset \\ C(a_i, a_j) \times \\ \prod_{k \in K(a_i, a_j)} \dfrac{1 - d_k(a_i, a_j)}{1 - C(a_i, a_j)}, \\ if\ K(a_i, a_j) \neq \emptyset, \end{cases} \tag{6}$$

where $K(a_i, a_j)$ contains the criteria such that $d_k(a_i, a_j) > C(a_i, a_j)$.

Equation 6 operates under the idea that if the magnitude of the concordance exceeds that of the discordance, there is no need to alter the concordance value. However, if this condition is not met, we must question the assertion $a_i S a_j$ and adjust the value of $C(a_i, a_j)$ accordingly.

In the scenario where the discordance value is 1.0 for any $(a_i, a_j) \in A \times A$ on any criterion $k$, there is no assurance that $a_i S a_j$, hence the outranking degree is $\sigma(a_i, a_j) = 0.0$.

Based on Eq. 6, we can create a fuzzy outranking relation $O_A^\sigma$ stated on $A \times A$ where any $(a_i, a_j) \in A \times A$ has a value $\sigma(a_i, a_j), (0 \leq \sigma(a_i, a_j) \leq 1)$ indicating the power of $a_i O a_j$.Once the model is complete, the subsequent phase in the approach involves exploiting the outranking model $O_A^\sigma$ to generate a ranking of alternatives. ELECTRE III employs the distillation algorithm [17] to exploit the outranking model $O_A^\sigma$ to produce a ranking.

However, due to space limitations, we will not elaborate on the details of this procedure here.

**Table 5.** Concordance indices on the criterion $g_1$

| | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_6$ | $ei_7$ |
|---|---|---|---|---|---|---|---|
| $ei_1$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_4^P, 0)$ | $(S_8^P, 0)$ |
| $ei_2$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_3$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_4$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_5$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_6$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_7$ | $(S_8^P, 0)$ | $(S_4^P, 0)$ | $(S_4^P, 0)$ | $(S_8^P, 0)$ | $(S_4^P, 0)$ | $(S_4^P, 0)$ | $(S_8^P, 0)$ |

**Table 6.** Comprehensive linguistic concordance matrix $\bar{C}(ei_i, ei_j)$

| $\bar{C}(ei_i, ei_j)$ | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_6$ | $ei_7$ |
|---|---|---|---|---|---|---|---|
| $ei_1$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.584)$ | $(S_7^P, 0.84)$ | $(S_7^P, 0.584)$ | $(S_5^P, 0.8048)$ | $(S_7^P, 0.2448)$ |
| $ei_2$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.744)$ | $(S_7^P, 0.84)$ | $(S_7^P, 0.744)$ | $(S_7^P, 0.4048)$ | $(S_7^P, 0.2448)$ |
| $ei_3$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_4$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.488)$ | $(S_8^P, 0)$ | $(S_7^P, 0.488)$ | $(S_7^P, 0.3088)$ | $(S_7^P, 0.6288)$ |
| $ei_5$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.7776)$ | $(S_7^P, 0.7776)$ |
| $ei_6$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_7$ | $(S_8^P, 0)$ | $(S_6^P, 0.24)$ | $(S_6^P, 0.24)$ | $(S_8^P, 0)$ | $(S_6^P, 0.24)$ | $(S_4^P, 0.8)$ | $(S_8^P, 0)$ |

Instead, in the following subsections, we introduce basic concepts of computers with words.

### 2.2 Linguistic Information and Management of Heterogeneous Information in MCDA

This section provides an overview of approaches to handling the three types of information in the heterogeneous framework. It introduces the 2-tuple linguistic representation model, which is appropriate for our problem because it enhances the interpretability of the MCDA process, which are the main required features of our proposal.

#### 2.2.1 The Heterogeneous Framework

Here, the evaluation framework calculates a global evaluation that condenses the gathered information and gives helpful decision-making results.

The DM can naturally declare his preferences in different information domains and obtain a heterogeneous structure [18]. The following expression domains are used in the linguistic extension of the ELECTRE III method:

- Numerical values $(N)$: $g_j(a_i) = v_{ij} \in [a, b], a, b \in \Re$ Represents assessments related to quantitative criteria.
- Interval values $(V)$: $g_j(a_i) = V([a, b]) = [a_{ij}, b_{ij}] \in [a, b], \quad and \quad a_{ij} \leq b_{ij}$. When exact numbers are unavailable, decision-makers use imprecise quantitative criteria.
- Linguistic values $(S)$: $g_j(a_i) = s_{ij} \in S, \quad S = \{s_0, \ldots, s_h\}$, being $h + 1$ the number of elements of the linguistic term set (LTS) $S$.

Assessing qualitative criteria is familiar to them. The linguistic approach is appropriate for representing data through linguistic variables. [19, 20].

#### 2.2.2 The 2-tuple Linguistic Representation Model

Handling heterogeneous information can be done using processes based on computing with words [20]. The models most frequently used for the treatment of heterogeneous information are:

i. The semantic model utilizes linguistic terms as labels to represent fuzzy numbers, while the computations are performed directly on the fuzzy numbers.
ii. The symbolic model that utilizes an order index of the linguistic terms to perform direct calculations on the labels.

**Table 7**. Linguistic discordance matrix on a criterion $g_1$

| | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_6$ | $ei_7$ |
|---|---|---|---|---|---|---|---|
| $ei_1$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_2$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_3$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_4$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_5$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_6$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |
| $ei_7$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ | $(S_0^P, 0)$ |

**Table 8**. Linguistic credibility matrix in the linguistic extension of the ELECTRE III

| $\bar{C}(ei_i, ei_j)$ | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_6$ | $ei_7$ |
|---|---|---|---|---|---|---|---|
| $ei_1$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.584)$ | $(S_7^P, 0.84)$ | $(S_7^P, 0.584)$ | $(S_5^P, 0.8048)$ | $(S_7^P, 0.2448)$ |
| $ei_2$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.744)$ | $(S_7^P, 0.84)$ | $(S_7^P, 0.744)$ | $(S_7^P, 0.4048)$ | $(S_7^P, 0.2448)$ |
| $ei_3$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_4$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.488)$ | $(S_8^P, 0)$ | $(S_7^P, 0.488)$ | $(S_7^P, 0.3088)$ | $(S_7^P, 0.6288)$ |
| $ei_5$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_7^P, 0.7776)$ | $(S_7^P, 0.7776)$ |
| $ei_6$ | $(S_8^P, 0)$ | $(S_7^P, 0.68)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ | $(S_8^P, 0)$ |
| $ei_7$ | $(S_8^P, 0)$ | $(S_6^P, 0.24)$ | $(S_6^P, 0.24)$ | $(S_8^P, 0)$ | $(S_6^P, 0.24)$ | $(S_4^P, 0.8)$ | $(S_8^P, 0)$ |

This research uses the symbolic model to calculate the linguistic evaluations in the ELECTRE III method, using the 2-tuple linguistic representation model developed by [11]. In the rest of this section, we present the basics of the 2-tuple linguistic representation model.

**Definition 2.1.** [4]. Let $S = \{s_0, \ldots, s_h\}$ be a linguistic term set. The symbolic translation of a linguistic term $s_i \in S = \{s_0, \ldots, s_h\}$ is a numerical value $\alpha$ assessed in $[-0.5, 0.5)$ that supports the "difference of information" between an amount of information $\beta \in [0, h]$ and the closest value in $\{0, \ldots, h\}$ that specifies the index of the closest linguistic term in $S(s_i)$ being $[0, h]$ the interval of granularity of $S$.

Based on this meaning, a linguistic representation model must be built that denotes linguistic information using a 2-tuple $(s_i, \alpha_i)$, $s_i \in S$ means the linguistic label of information and $\alpha_i \in [-0.5, 0.5)$ is a numerical value stating the translation starting from the original result $\beta$ with the index nearest to the label, $i$, in the linguistic term set $S(s_i)$, i.e., the symbolic translation. Moreover, this model states transformation functions between the numerical values and the linguistic 2-tuple.

**Definition 2.2.** [4]. Let $S = \{s_0, \ldots, s_h\}$ a linguistic term set and $\beta \in [0, h]$ a value supporting the result of a symbolic aggregation operation, then the 2-tuple that states the equivalent information to $\beta$ is calculated with the following function:

$$\Delta_S: [0, h] \to S \times (-0.5, .0.5), \tag{7}$$

$$\Delta_S(\beta) = (s_i, \alpha), with \quad \begin{cases} s_i & i = round(\beta), \\ \alpha = \beta - i & \alpha \in [-0.5, 0.5), \end{cases} \tag{8}$$

where round $(\cdot)$ is the typical round operation, $S_i$ has the closest index label to $\beta$, and $\alpha$ is the value of the symbolic translation.

Let $S = \{s_0, \ldots, s_h\}$ be a linguistic term set and $(s_i, \alpha_i)$ be a linguistic 2-tuple. From (7) and (8), a ${\Delta_S}^{-1}$ function can be defined, so that, from a 2-tuple $(s_i, \alpha_i)$, ${\Delta_S}^{-1}$ returns its equivalent numerical value $\beta \in [0, h]$ in the interval of granularity of $S$ as follows:

$${\Delta_S}^{-1}: S \times -0{,}5{,}0.5) \to [0, h], \tag{9}$$

$${\Delta_S}^{-1}(s_i, \alpha) = i + \alpha = \beta. \tag{10}$$

Note that to transform a linguistic term into a linguistic 2-tuple, append a value 0 as symbolic translation: $s_i \in S \Rightarrow (s_i, 0)$.

**Example 1**. Let us suppose a symbolic aggregation operator, φ(.) whose input are different labels assessed in *S* = {*nothing*; *very low*; *low*; *medium*; *high*; *very high*; *perfect*}, obtaining the following results:

**Table 9**. Crisp outranking relation

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_3$ |
|---|---|---|---|
| $ei_1$ | 0 | 0 | 0 |
| $ei_2$ | 0 | 0 | 0 |
| $ei_3$ | 1 | 0 | 0 |
| $ei_4$ | 1 | 1 | 0 |
| $ei_5$ | 1 | 0 | 0 |
| $ei_6$ | 1 | 0 | 0 |
| $ei_7$ | 1 | 0 | 0 |

φ (medium; medium; medium; very high) = 3:21 = $\beta_1$

φ (low; medium; very low; high) = 2:76 = $\beta_2$

Being $\beta_1$=3:21 and $\beta_2$=2:76, then the 2-tuple linguistic values (Definition 2.2) of these symbolic results, which do not match with any linguistic term in *S,* are:

$\Delta_S$(3:21) = ($s_3$;0:21) = (*medium*, 0.21).

The symbolic translation (definition 2.1) $\alpha$ is 0.21.

$\Delta_S$(1:75) = ($s_3$;- 0:24) = (*medium*,-0.24). The symbolic translation $\alpha$ is -0.24.

### 2.2.3 Aggregation of 2-tuples

This process involves obtaining a single value representing a set of values of the same type; therefore, adding a series of linguistic 2-tuples must be a linguistic 2-tuple.

In the literature, we can find various 2-tuple aggregation operators (e.g., [4]) based on the classical aggregation operators, such as the arithmetic mean and weighted mean operators.

**Definition 2.3.** Let $x = \{(s_1, \alpha_1), \ldots, (s_n, \alpha_n)\}$ be a set of 2-tuples; the extended Arithmetic Mean $AM^*$ using the linguistic 2-tuples is computed as:

$$\begin{aligned} &AM^*\big((s_1, \alpha_1), \ldots, (s_n, \alpha_n)\big), \\ &= \Delta\left(\sum_{i=1}^{n} \frac{1}{n} \Delta^{-1}(s_i, \alpha_i)\right), \\ &= \Delta\left(\frac{1}{n}\sum_{i=1}^{n} \beta_i\right). \end{aligned} \tag{11}$$

**Definition 2.4.** Let $\{(s_1, \alpha_1) \ldots (s_n, \alpha_n)\}$ be a set of linguistic 2-tuples, and $W = \{w_1, \ldots, w_n\}$ the set of its associated weights. Then, the 2-tuple weighted mean, $W\_AM^*$, is computed as:

$$\begin{aligned} &W_{AM^*((s_1,\alpha_1)}, \ldots., (s_n, \alpha_n)\big), \\ &= \Delta\left[\frac{\sum_{i=1}^{n} \Delta^{-1}(s_i, \alpha_i) w_i}{\sum_{i=1}^{n} w_i}\right], \\ &= \Delta\left[\frac{\sum_{i=1}^{n} \beta_i . w_i}{\sum_{i=1}^{n} w_i}\right]. \end{aligned} \tag{12}$$

### 2.2.4 Comparison of 2-tuples

The 2-tuple information is compared using the lexicographic order. Let $(s_k, \alpha_1)$ and $(s_l, \alpha_2)$ be two 2-tuples represented by two assessments:

- If $k < I$ then $(s_k, \alpha_1)$ is smaller than $(s_l, \alpha_2)$
- If $k = I$ then:
  1. If $\alpha_1 = \alpha_2$, then $(s_k, \alpha_1)$ and $(s_l, \alpha_2)$ represent the same value.
  2. If $\alpha_1 < \alpha_2$, then $(s_k, \alpha_1)$ is smaller than $(s_l, \alpha_2)$.
  3. If $\alpha_1 > \alpha_2$, then $(s_k, \alpha_1)$ is bigger than $(s_l, \alpha_2)$.

### 2.2.5 Negation Operator of a 2-tuple

The negation operator over 2-tuples is expressed as:

$$Neg(s_i, \alpha) = \Delta_S(h - \Delta_S{}^{-1}(s_i, \alpha)), \tag{13}$$

where $h + 1$ is the number of elements in $S = \{s_0, \ldots, s_h\}$, $s_i \in S$.

## 2.3 The Linguistic Fusion Approach for Heterogeneous Information based on the 2-tuple Fuzzy Linguistic Model

The approach used in this section to fuse heterogeneous information based on the 2-tuple linguistic model considers various transformation functions that go from numerical, interval, and linguistic information sources toward a common linguistic format [21].

1 Choosing the basic linguistic term set (BLTS) $S_{\mathrm{BLTS}} = \{s_0, s_1, \ldots, s_h\}$: The BLTS must have the maximum granularity to maintain the uncertainty degree associated with the DM and the capacity of discrimination to express the preference values [9].

2 Transformation of the heterogeneous information into fuzzy sets in a linguistic domain:

**Table 10.** Power, weakness, and qualification scores

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_6$ | $ei_7$ |
|---|---|---|---|---|---|---|---|
| $p_{D_0}^{(s_c,\alpha_c)^{(0)}}$ - power | 0 | 0 | 3 | 2 | 2 | 4 | 2 |
| $f_D^{(s_c,\alpha_c)^{(0)}}$ - weakness | 5 | 1 | 0 | 4 | 1 | 0 | 2 |
| $q_{D_0}^{(s_c,\alpha_c)^{(0)}}$ - qualification | -5 | -1 | 3 | -2 | 1 | 4 | 0 |

Each input value $x$ is transformed into a fuzzy set on $S_{BLTS}$, $F(S_{BLTS})$ by means of one of the following transformation functions: For $x \in [a,b]$, the numerical transformation function $T_{NS_{BLTS}}: [a,b] \to F(S_{BLTS})$ is expressed as:

$$T_{NS_{BLTS}}(x) = \sum_{i=0}^{h} (S_i/\lambda_i), \tag{14}$$

where $\lambda_i = \mu_{s_i}(x) \in [0,1]$ is the membership degree of x to $s_i \in S_{BLTS}$:

$$\mu_{s_i}(x) = \begin{cases} 0 & if \quad x \notin Support\ \mu_{s_i}(x) \\ \dfrac{x-a_i}{b_i-a_i} & if \quad a_i \le x \le b_i, \\ 1 & if \quad b_i \le x \le d_i, \\ \dfrac{c_i-x}{c_i-d_i} & if \quad d_i \le x \le c_i. \end{cases} \tag{15}$$

For $x \in V([a,b])$, the interval transformation function $T_{VS_{BLTS}}: V([a,b]) \to F(S_{BLTS})$ is expressed as:

$$T_{VS_{BLTS}}(x) = \sum_{i=0}^{h} (S_i/\lambda_i), \tag{16}$$

where:

$$\lambda_i = \max_y \min\{\mu_V(y), \mu_{s_j}(y)\}, \tag{17}$$

$$i \in \{0,\dots,h\}, \tag{18}$$

$$\mu_V(y) = \begin{cases} 0 & if \quad y < a, \\ 1 & if \quad a \le y \le b, \\ 0 & if \quad y > b. \end{cases} \tag{19}$$

For $x = s_j \in S$ with , $S = \{S_0, \dots., S_h\}$ the linguistic transformation function $T_{SS_{BLTS}}: S \to F(S_{BLTS})$ is expressed as:

$$T_{SS_{BLTS}}(S_i) = \sum_{i=0}^{h} (S_i/\lambda_i), \tag{19}$$

where:

$$\lambda_i = \max_y \min\{\mu_{s_j}(y), \mu_{s_i}(y)\}, \quad i \in \{0,\dots,h\}. \tag{20}$$

This information fusion process [13] is illustrated in Figure 1.

### 2.4 Transformation of Fuzzy Sets into Linguistic 2-tuple Values

Here, the fuzzy sets are converted into linguistic 2-tuples over the BLTS through the function $\chi: F(S_{BLTS}) \to S \times [-0.5, 0.5)$, which is stated as follows:

$$\chi(\lambda_0, \lambda_1, \dots, \lambda_h) = \Delta_S\left(\frac{\sum_{i=0}^{h} i\lambda_i}{\sum_{i=0}^{h} \lambda_i}\right) = (s,\alpha) = \bar{s} \in \bar{S}_{BLTS}, \tag{21}$$

where $S = \{S_0, S_1, \dots., S_h\}$ is the set of linguistic terms, and $\bar{S} = S \times [-0.5, 0.5)$ is the linked 2-tuple term set. The function $\Delta_S$ is defined in section 2.2.2. After the transformations of heterogeneous information into 2-tuple linguistic values in the BLTS have been carried out, we can use the 2-tuple linguistic computation model [7] to compute linguistic results in $\bar{S}_{BLTS}$. This step uses 2-tuple linguistic aggregation operators [20, 22]. Based on the results presented in this section, we present a linguistic extension of the ELECTRE III method in the following section.

## 3 The Linguistic ELECTRE III Method

This section presents our proposal for the linguistic extension of the ELECTRE III model; to this end, a procedure is defined to model the partial and global concordance indices, the discordance indices by criterion, the thresholds of the criteria, and the credibility index linguistically so that they can accept 2-tuple linguistic values.

Consequently, the linguistic ELECTRE III method provides a more realistic operability of the qualitative criteria when solving multicriteria ranking problems.

Within this procedure, a linguistic difference function allows for the calculation of the linguistic difference for each pair of alternatives for each

**Table 11.** Crisp outranking relation for Iteration 1 in Distillation 2

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_7$ |
|---|---|---|---|---|---|---|
| $ei_1$ | 0 | 0 | 0 | 0 | 0 | 0 |
| $ei_2$ | 0 | 0 | 0 | 0 | 0 | 0 |
| $ei_3$ | 1 | 0 | 0 | 1 | 0 | 1 |
| $ei_4$ | 1 | 1 | 0 | 0 | 0 | 0 |
| $ei_5$ | 1 | 0 | 0 | 1 | 0 | 0 |
| $ei_7$ | 1 | 0 | 0 | 1 | 0 | 0 |

decision criterion. The linguistic output provided by the linguistic difference function serves as the linguistic input of the linguistic concordance and discordance indices for each criterion.

The output of the concordance and discordance indices are expressed in the same linguistic scale to preserve interpretability. Figure 2 schematically presents the process for modeling the linguistic outranking index in three phases.

### 3.1 Fusion of Heterogeneous Information

In a multicriteria ranking problem with a heterogeneous information environment, alternatives are evaluated using diverse expression domains based on the uncertainty and criteria type, as well as each DM's experience.

#### 3.1.1 Transformation into Fuzzy Sets

The expression domains (numerical, interval, and linguistic) used in the heterogeneous framework are presented in this part of the first phase. The fusion approach handles these three types of information.

Previously, a unification domain $S_{BLTS}$ is defined as allowing the transforming of heterogeneous information by fuzzy sets into $S_{BLTS}$, using the respective transformation functions of Eqs. 14, 16, and 19.

#### 3.1.2 Transformation into 2-tuples Linguistic Values

Then, the process transforms the fuzzy sets into 2-tuple linguistic values in $S_{BLTS}$ using Eq. (21). Hence, the fused evaluation for each criterion $g_k$ concerning each alternative $a_i$, is represented in a 2-tuple linguistic value $\bar{g}_k(a_i) = (s_i, \alpha_i) \in S_{BLTS}$.

### 3.2 The Linguistic Difference Function $D_S$

This function is introduced to facilitate the computation of the linguistic concordance and discordance indices because the input values of the indices and thresholds must be linguistic values for a correct interpretation.

To compare two linguistic values, we need a comparison scale that can measure the linguistic difference between them. The scale's granularity will depend on the decision maker's knowledge, who needs to interpret the difference between the two alternatives using a bipolar scale [23]. This type of scale is convenient because it has a neutral point, which separates the positive differences from the negative ones [5].

In short, this function's linguistic output is the input of the linguistic concordance and discordance indices. The linguistic difference function is expressed in the linguistic scale, and the threshold parameters are stated accordingly. Consequently, a proper linguistic difference function between linguistic preference values is necessary for developing an extension of ELECTRE III dealing with fuzzy linguistic information.

**Definition 3.1**. Let $S_{BLTS} = \{s_0, \ldots, s_h\}$ and $S^C = \{l_0^C, \ldots, l_{\hat{h}}^C\}$ be the set of linguistic terms for preference values and the set of linguistic terms to express the linguistic difference value between two terms in $S_{BLTS}$, respectively. Let $(s_i, \alpha_i)$ and $(s_j, \alpha_j)$ be two 2-tuple linguistic values stated in $S_{BLTS}$. The linguistic difference value between $(s_i, \alpha_i)$ and $(s_j, \alpha_j)$ expressed in $\bar{S}^C$ is calculated by:

$$D_S: \bar{S}_{BLTS} \times \bar{S}_{BLTS} \rightarrow \bar{S}^C, \tag{22}$$

$$D_S\left((s_i, \alpha_i), (s_j, \alpha_j)\right) = \Delta_{s^c}\left(\frac{\left(\left(\Delta^{-1}_{\bar{S}_{BLTS}}(s_j, \alpha_j) - \Delta^{-1}_{\bar{S}_{BLTS}}(s_i, \alpha_i)\right) + h\right)}{2.h}.\hat{h}\right). \tag{23}$$

The proposed linguistic difference function satisfies the following properties:

– The difference between the same value of $S_{BLTS}$ is the neutral point value of $S^C$:

$$D_s\big((s_i, \alpha_i), (s_i, \alpha_i)\big) = neutral\ point\{S^c\} = l^c_{\hat{h}/2}. \tag{24}$$

– The difference between the minimum (maximum) and maximum (minimum) values of $S_{BLTS}$ must be the maximum (minimum) value of $S^C$:

$$D_s\big((s_0,0),(s_h,0)\big) = max\{S^c\} = l_h^c,$$
$$D_s\big((s_h,0),(s_0,0)\big) = min\{S^c\} = l_0^c. \quad (25)$$

The proof of these properties is trivial. We propose the following syntax for $S^C$:

$$\begin{gathered} S^c = \{l_0^c: Extremely\_Lower(EL),\\ l_1^c: Much_Lower(ML),\\ l_2^c: Lower(L), l_3^c: Slightly\_Lower(SL),\\ l_4^c: Identical(I), l_5^c: Slightly_{Higher(SH)},\\ l_6^c: Higher(H),\\ l_7^c: Much\_Higher(MH),\\ l_8^c: Extremely\_Higher(EH)\}. \end{gathered} \quad (26)$$

**Example 2**. In this example, we perform the linguistic difference of the 2-tuple linguistic values $(L,-0.2)$ and $(H,-0{,}3)$ using the linguistic difference function (Definition 3.1). The set of linguistic terms for preference values is defined as follows:

$$\begin{gathered} S_{BLTS}\\ = \{s_0: Very\_low(VL),\ s_1: Low(L),\ s_2: Medium(M),\\ s_3: High(H),\ s_4 Very\_high(VH)\}. \end{gathered} \quad (27)$$

With the linguistic terms $S^C$ to describe the difference function defined above:

$$\begin{gathered} D_s\big((L,-0.2),(H,-0.3)\big) = \Delta_{s^c}\left(\frac{((2.7-0.8)+4)}{(2).(4)}.8\right)\\ = \Delta_{s^c}(5.9) = (l_5^c, 0.9) = (SH, 0.9). \end{gathered} \quad (28)$$

### 3.3 Linguistic Concordance and Discordance Indices

The linguistic concordance and discordance indices defined in this section have 2-tuple linguistic values as input and output.

#### 3.3.1 The Linguistic Concordance Index

The linguistic concordance value of $(a_i,a_j) \in A \times A$ for a criterion $k$ is calculated by making use of a linguistic concordance index $\bar{C}_k(a_i,a_j)$ through the linguistic difference function. The input value of the $k-th$ concordance index is a linguistic difference value of:

**Table 12.** Power, weakness, and qualification scores for Iteration 1 in Distillation 2

| $ei_i Oei_j$ | $ei_1$ | $ei_2$ | $ei_3$ | $ei_4$ | $ei_5$ | $ei_7$ |
|---|---|---|---|---|---|---|
| $p_{D_0}^{(s_{l_c},\alpha_c)^{(0)}}$ - power | 0 | 0 | 3 | 2 | 2 | 2 |
| $f_D^{(s_{l_c},\alpha_c)^{(0)}}$ - weakness | 4 | 1 | 0 | 3 | 0 | 2 |
| $q_{D_0}^{(s_{l_c},\alpha_c)^{(0)}}$ - qualification | -4 | -1 | 3 | -1 | 2 | 0 |

$$\begin{gathered} (a_i,a_j) \in A \times A\\ D_s\left(\bar{g}_k(a_i), \bar{g}_k\big(a_j\big)\right) \in \bar{S}^c, \end{gathered} \quad (29)$$

where $g_k: A \to S_{BLTS}$ is the $k-th$ criteria function. The output value of the linguistic concordance function, i.e., the concordance value, is likewise represented by a value in a linguistic concordance scale $\bar{S}^P$.

$S^P = \{s_0^P, \ldots, s_{h_P}^P\}$ represents a linguistic term set. The granularity of $S^P$ is chosen following the DM's knowledge to make clear the concordance value of $(a_i,a_j) \in A \times A$.

Here, $(s_0^P, 0)$ represents no concordance and $(s_{h_P}^P, 0)$ strict concordance.

The concordance index $\bar{C}_k(a_i,a_j)$ varies from $(s_0^P,0)$ to $(s_{hp}^P,0)$. If $\bar{C}_k(a_i,a_j)$ is equal to $(s_0^P,0)$, then $a_i$ is worse than $a_j$.

The indifference and preference thresholds $\left(l_{t_{q_k}}, \alpha_{q_k}\right)_q$ and $\left(l_{t_{p_k}}, \alpha_{p_k}\right)_p$ respectively, both in $\bar{S}^C$, are used to construct a concordance index $\bar{C}_k(a_i,a_j)$ for each criterion $k$ , defined by:

**Definition 3.2.** *The linguistic concordance index concerning a criterion $g_k$, $\bar{C}_k(a_i,a_j)$, that symbolizes the linguistic concordance value stated in 2-tuple linguistic values in $S^P = \{s_0^P, \ldots, s_{h_P}^P\}$ of the linguistic difference value between $a_j$ over $a_i$, regarding criterion $k$, $\bar{D}_s\big(a_i,a_j\big)_k = D_s\left(\bar{g}_k(a_i), \bar{g}_k\big(a_j\big)\right) \in \bar{S}^c$ is specified as:*

$$\bar{C}_k(a_i,a_j): A \times A \to \bar{S}^P, \quad (30)$$

**Table 13.** Crisp outranking relation for Iteration 1 in Distillation 3

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_4$ | $ei_5$ | $ei_7$ |
|---|---|---|---|---|---|
| $ei_1$ | 0 | 0 | 0 | 0 | 0 |
| $ei_2$ | 0 | 0 | 0 | 0 | 0 |
| $ei_4$ | 1 | 1 | 0 | 0 | 0 |
| $ei_5$ | 1 | 0 | 1 | 0 | 0 |
| $ei_7$ | 1 | 0 | 1 | 0 | 0 |

$$\bar{C}_k(a_i,a_j) = \begin{cases} (S_0^P,0), & if \quad \bar{D}_s(a_i,a_j)_k > \left(l_{t_{p_k}},\alpha_{p_k}\right)_p \\ \Delta_{s^p}\left(\dfrac{\Delta_{s_c}^{-1}\left(l_{t_{p_k}},\alpha_{p_k}\right)_p - \frac{\bar{h}}{2} - \left[\Delta_{s_c}^{-1}\left(\bar{D}_s(a_i,a_j\right.\right.}{\Delta_{s_c}^{-1}\left(l_{t_{p_k}},\alpha_{p_k}\right)_p - \frac{\bar{h}}{2}\left(\ \Delta_{s_c}^{-1}(l_{t_{q_k}},\alpha_{q_k}\right.}\right) & \\ if \ \left(l_{t_{q_k}},\alpha_{q_k}\right)_q < \bar{D}_s(a_i,a_j)_k \leq \left(l_{t_p}\right. & \\ \left(S_{h_p}^P,0\right), \quad if \quad \bar{D}_s(a_i,a_j)_k \leq \left(l_{t_{q_k}},\alpha_{q_k}\right) & \end{cases} \tag{31}$$

With $k = 1, \ldots, n$.

The concordance index $\bar{C}_k(a_i,a_j)$ is a linguistic index measuring whether "$a_i$ is at least as good as $a_j$" on criterion $k$.

### 3.3.2 The Linguistic Discordance Index

For each criterion $g_k$ a linguistic discordance index $\bar{d}_k(a_i,a_j)$ can be defined. This index measures how much $g_k$ is more or less discordant with the affirmation "$a_i$ outranks $a_j$." This index considers a linguistic veto threshold $\left(l_{t_{v_k}},\alpha_{v_k}\right)_v$ to calculate linguistic concordance.

It should be mentioned that any outranking of $a_i$ by $a_j$ by specified by the concordance index can be overruled if there is any criterion $g_k$ for which the alternative $a_j$ outperforms the alternative $a_i$ by at least a veto threshold, even if all the other criteria favor the outranking of $a_i$ $(\bar{D}_s(a_i,a_j)_k \geq \left(l_{t_{v_k}},\alpha_{v_k}\right))$.

So, if $a_i$ is better than $a_j$ normally, there may be some criteria (possibly one) where $a_i$ is worse than $a_j$. The index $\bar{d}_k(a_i,a_j)$ displays this condition for that criterion. $\bar{d}_k(a_i,a_j)$ varies from $(s_0^P,0)$ to $(s_{h_P}^P,0)$. $(s_0^P,0)$ represents no discordance and $(s_{h_P}^P,0)$ represents strict discordance. The linguistic discordance index is calculated according to the following definition:

**Definition 3.3**. *The linguistic discordance index $\bar{d}_k(a_i,a_j)$, for a criterion $g_k$, that represents the linguistic discordance value expressed in 2-tuple linguistic values in $S^P = \{s_0^P,\ldots,s_{h_P}^P\}$ of the linguistic difference value between $a_j$ over $a_i$, regarding criterion $k$, $\bar{D}_S(a_i,a_j)_k = D_S(\bar{g}_k(a_i),\bar{g}_k(a_j)) \in \bar{S}^C$ is stated as:*

$$\bar{d}_k(a_i,a_j): A \times A \to \bar{S}^P, \tag{32}$$

$$\bar{d}_k(a_i,a_j) = \begin{cases} (S_0^P,0), & if \quad \bar{D}_s(a_i,a_j)_k \leq \left(l_{t_{p_k}},\alpha_{p_k}\right)_p \\ \Delta_{s^p}\left(\dfrac{\Delta_{s_c}^{-1}\left(\bar{D}_s(a_i,a_j)_k\right) - \frac{\bar{h}}{2} - \left[\Delta_{s_c}^{-1}\left(l_{t_{p_k}},\alpha_{p_k}\right)_p - \frac{\bar{h}}{2}\right]}{\Delta_{s_c}^{-1}\left(l_{t_{v_k}},\alpha_{v_k}\right)_v - \frac{\bar{h}}{2} - \left(\Delta_{s_c}^{-1}(l_{t_{p_k}},\alpha_{p_k})_p - \frac{\bar{h}}{2}\right)}.h_p\right), & \\ if \ \left(l_{t_{p_k}},\alpha_{p_k}\right)_p < \bar{D}_s(a_i,a_j)_k < \left(l_{t_{v_k}},\alpha_{v_k}\right)_v & \\ \left(S_{h_P}^P,0\right), \quad if \quad \bar{D}_s(a_i,a_j)_k \geq \left(l_{t_{v_k}},\alpha_{v_k}\right)_v & \end{cases} \tag{33}$$

with $k = 1, \ldots, n$.

## 3.4 The Linguistic Outranking Relation in the Linguistic ELECTRE III

The linguistic outranking relation $O_A^{\bar{\sigma}}$ defined for each $(a_i,a_j) \in A \times A$ as a linguistic credibility index, $\bar{\sigma}(a_i,a_j)$, state broadly in what linguistic measure "$a_i$ outranks $a_j$" employing both the linguistic concordance index $\bar{C}(a_i,a_j)$ and the linguistic discordance indices $\bar{d}_k(a_i,a_j)$ for each criterion $g_k$.

The linguistic credibility index is the comprehensive linguistic concordance index reduced by the linguistic discordance indices. In the nonappearance of such linguistic discordance criteria, $\bar{\sigma}(a_i,a_j) = \bar{C}(a_i,a_j)$.

This linguistic credibility value is decreased in the occurrence of one or more linguistic discordant criteria $g_k$ when $\bar{d}_k(a_i,a_j) > \bar{C}(a_i,a_j)$. In correspondence with the veto effect $\bar{\sigma}(a_i,a_j) = (s_0^P,0)$ if exists a linguistic discordance index such that $\bar{d}_k(a_i,a_j) = (s_{h_P}^P,0)$, does not matter what the weight of the criterion $w_k$ is. The linguistic credibility index $\bar{\sigma}(a_i,a_j)$ is defined as follows:

**Table 14.** Power, weakness, and qualification scores for Iteration 1 in Distillation 3

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_4$ | $ei_5$ | $ei_7$ |
|---|---|---|---|---|---|
| $p_{D_0}^{(s_{t_c},\alpha_c)^{(0)}}$ - power | 0 | 0 | 2 | 2 | 2 |
| $f_D^{(s_{t_c},\alpha_c)^{(0)}}$ - weakness | 3 | 1 | 2 | 0 | 0 |
| $q_{D_0}^{(s_{t_c},\alpha_c)^{(0)}}$ - qualification | -3 | -1 | 0 | 2 | 2 |

$$= \begin{cases} \bar{\sigma}(a_i,a_j) \\ \bar{C}(a_i,a_j), if\ \bar{K}(a_i,a_j)=\phi \\ \Delta_{S^P}(\Delta_{S^P}^{-1}(\bar{C}(a_i,a_j)) \bullet \\ \prod_{k\in\bar{K}(a_i,a_j)} \frac{\Delta_{S_P}^{-1}(s_{h_P}^P,0)-\Delta_{S_P}^{-1}\left(\bar{d}_k(a_i,a_j)\right)}{\Delta_{S_P}^{-1}(s_{h_P}^P,0)-\Delta_{S_P}^{-1}\left(\bar{C}(a_i,a_j)\right)}) \\ if\ \bar{K}(a_i,a_j)\neq\phi, \end{cases} \quad (34)$$

where:

$$\bar{K}(a_i,a_j)=\{g_k\in G\,|\,\bar{d}_k(a_i,a_j)>\bar{C}(a_i,a_j)\}. \quad (35)$$

The formula for determining the linguistic value of $\bar{\sigma}(a_i,a_j)$ over the linguistic interval $((s_0^P,0),(s_{h_P}^P,0))$ is non-compensatory, i.e., an alternative's notable poor performances in some criteria cannot be compensated for even with very high performance in other criteria. The aggregated performance exposes this fact. This completes the first phase of the linguistic ELECTRE III method.

### 3.5 The Ranking Algorithm in the Linguistic Extension of ELECTRE III

The second phase of the linguistic ELECTRE method is to exploit the linguistic outranking relation $O_A^{\bar{\sigma}}$ to get a partial preorder of the alternatives. This final partial preorder is obtained because of the "intersection" of two complete preorders resulting from the descending and ascending distillations [24].

In the descending distillation, the procedure ranks the alternatives from the best to the worst; on the contrary, in the ascending distillation, the process ranks the alternatives from the worst to the best.

In the following, we modify the distillation procedure of ELECTRE III. In the linguistic ELECTRE III distillation procedure, we state a set of linguistic credibility cutting levels $(s_{t_c},\alpha_c)^{(r)}$ in $S^P=\{s_0^P,\ldots,s_{h_P}^P\}$. Given a *linguistic cutoff* level symbolized by $(s_{t_c},\alpha_c)^{(r)}$, both distillations relate to the following linguistic crisp outranking relation:

$$a_i O_A^{(s_{t_c},\alpha_c)^{(r)}} a_j \quad (36)$$

$$\Leftrightarrow \begin{cases} \bar{\sigma}(a_i,a_j)\geq(s_{t_c},\alpha_c)^{(r)} \\ \bar{\sigma}(a_i,a_j)>\Delta_{S^P}\left(\Delta_{S^P}^{-1}(\bar{\sigma}(a_j,a_i))+z(\bar{\sigma}(a_i,a_j))\right), \end{cases}$$

where $z(\bar{\lambda})=\frac{\alpha(\Delta_{S^P}^{-1}(\bar{\lambda}))}{h_P}+\beta$ is a linguistic distillation threshold and $\alpha$ and $\beta$ are two distillation coefficients. $(s_{t_c},\alpha_c)^{(r)}$ is also a linguistic preference parameter, which fixes the minimum degree of credibility considered obligatory by the DM to support the statement "$a_i$ outranks $a_j$". From the linguistic crisp outranking relation, for each alternative $a_i$, its $(s_{t_c},\alpha_c)^{(r)}$-qualification is:

$$q_A^{(s_{t_c},\alpha_c)^{(r)}}(a_i)=p_A^{(s_{t_c},\alpha_c)^{(r)}}(a_i) - f_A^{(s_{t_c},\alpha_c)^{(r)}}(a_i), \quad (37)$$

where $p_A^{(s_{t_c},\alpha_c)^{(r)}}(a_i)=\left|\left\{a_j\in A: a_i O_A^{(s_{t_c},\alpha_c)^{(r)}} a_j\right\}\right|$ is the $(s_{t_c},\alpha_c)^{(r)}$-power of $a_i$; it is the number of alternatives that are outranked by $a_i$, and $f_A^{(s_{t_c},\alpha_c)^{(r)}}(a_i)=\left|\{a_j\in A: a_j O_A^{(s_{t_c},\alpha_c)^{(r)}} a_i\}\right|$ is the $(s_{t_c},\alpha_c)^{(r)}$-weakness of $a_i$; it is the number of alternatives outranking $a_i$.

In the rest of this section, we explain the descending and ascending distillation algorithms in detail as follows: Let $(s_{t_c},\alpha_c)^{(1)}$ be the first fixed *linguistic cutoff* level and $q_A^{(s_{t_c},\alpha_c)^{(1)}}(a_i)$ be the qualification of alternative $a_i$. Then, choose in $A$ the best ones resulting a subset of alternatives from $A$ that has the maximum qualification (descending selection, $\vec{D}_1$) or the worst alternatives resulting thus a subset of alternatives from $A$, which has the minimum qualification (ascending selection, $\overleftarrow{D}_1$):

**Table 15.** Crisp outranking relation for Iteration 2 in distillation 3

| $ei_iOei_j$ | $ei_5$ | $ei_7$ |
|---|---|---|
| $ei_5$ | 0 | 1 |
| $ei_7$ | 0 | 0 |

$$\vec{D}_1 = \left\{ \begin{matrix} a_i \in A | q_A^{(s_{t_c},\alpha_c)^{(1)}}(a_i) = \vec{q}_A \\ = \max\limits_{x \in A} q_A^{(s_{t_c},\alpha_c)^{(1)}}(x) \end{matrix} \right\}, \tag{38}$$

$$\overleftarrow{D}_1 = \left\{ a_i \in A \left| \begin{matrix} q_A^{(s_{t_c},\alpha_c)^{(1)}}(a_i) = \overleftarrow{q}_A \\ = \min\limits_{x \in A} q_A^{(s_{t_c},\alpha_c)^{(1)}}(x) \end{matrix} \right. \right\}. \tag{39}$$

Consequently, at the end of the $k$ steps of the first distillation, the first subset of $A$ is obtained, representing the first (last) class of one of the two final preorders. Let $\vec{C}_1 = \vec{D}_k$ symbolize the first class of the descending selection, and $\overleftarrow{C}_1 = \overleftarrow{D}_k$ indicate the last class of the ascending selection.

Let $\vec{A}_1 = A \backslash \vec{C}_1$, or $\overleftarrow{A}_1 = A \backslash \overleftarrow{C}_1$ represent the remaining subset of the alternatives from *A* to rank after the first distillation. In $\vec{A}_1$ and $\overleftarrow{A}_1$ The alternatives' qualification is computed again for choosing one or various alternatives. This process is reiterated until all the alternatives are ranked.

The distillation process is condensed in the following way:

1. Set $n = 0$, put or $\vec{A}_0 = A$ (descending), or $\overleftarrow{A}_0 = A$ (ascending).
2. Set: $(s_{t_c},\alpha_c)^{(0)} = \max\limits_{\substack{a_i,a_j \in \vec{A}_n \\ a_i \neq a_j}} \bar{\sigma}(a_i,a_j)$ *or* $(s_{t_c},\alpha_c)^{(0)} = \max\limits_{\substack{a_i,a_j \in \overleftarrow{A}_n \\ a_i \neq a_j}} \bar{\sigma}(a_i,a_j)$.
3. Put $k = 0,\ D_0 = \vec{A}_n$ (descending) or $D_0 = \overleftarrow{A}_n$ (ascending).
4. Choose the maximum value from the linguistic credibility scores that are less than $(s_{t_c},\alpha_c)^{(k)} - z((s_{t_c},\alpha_c)^{(k)})$.
5. $(s_{t_c},\alpha_c)^{(k+1)} = \max\limits_{\substack{\{\bar{\sigma}(a_i,a_j)+z((s_{t_c},\alpha_c)^{(k)})<(s_{t_c},\alpha_c)^{(k)}\} \\ a_i,a_j \in D_k}} \bar{\sigma}(a_i,a_j)$.
6. $If\ \forall a_i,a_j \in D_k,\ \bar{\sigma}(a_i,a_j) + z((s_{t_c},\alpha_c)^{(k)}) > (s_{t_c},\alpha_c)^{(k)},\ \ put\ \ (s_{t_c},\alpha_c)^{(k+1)} = (s_0,0)$.
7. Calculate the $(s_{t_c},\alpha_c)^{(k+1)}$-qualifications $(q_A^{(s_{t_c},\alpha_c)^{(k)}}(a_i)) \forall a_i \in D_k$.
8. Obtain the maximum or minimum $(s_{t_c},\alpha_c)^{(k+1)}$-qualification score: $\vec{q}_{D_k} = \max\limits_{x \in D_k} q_{D_k}^{(s_{t_c},\alpha_c)^{(k)}}(x)$ (descending) or $\overleftarrow{q}_{D_k} = \min\limits_{x \in D_k} q_{D_k}^{(s_{t_c},\alpha_c)^{(k)}}(x)$ (ascending).
9. Construct $\vec{D}_{k+1} = \left\{ a_i \in D_k \left| q_{D_k}^{(s_{t_c},\alpha_c)^{(k+1)}}(a_i) = \vec{q}_{D_k} \right. \right\}$ (descending) or $\overleftarrow{D}_{k+1} = \left\{ a_i \in D_k \left| q_{D_k}^{(s_{t_c},\alpha_c)^{(k+1)}}(a_i) = \overleftarrow{q}_{D_k} \right. \right\}$ (ascending).
10. If $\left|\vec{D}_{k+1}\right| = 1\ \ or\ \ \left|\overleftarrow{D}_{k+1}\right| = 1\ \ or\ \ (s_{t_c},\alpha_c)^{(k+1)} = (s_0,0)$ you proceed to step (9).
11. else, do $k = k+1,\ \ D_k = \vec{D}_k$ (descending) or $D_k = \overleftarrow{D}_k$ (ascending) and go to step (4).
12. $\vec{C}_{n+1} = \vec{D}_{k+1}$is the set of alternatives carried through the $(n+1)-th$ downward distillation, termed the $(n+1)-th$ distillate of the downward procedure. $\overleftarrow{C}_{n+1} = \overleftarrow{D}_{k+1}$ is the set of alternatives taken through the $(n+1)-th$ upward distillation, termed the $(n+1)-th$ distillate of the upward procedure.
13. Put $\vec{A}_{n+1} = \vec{A}_n \backslash \vec{C}_{n+1}$ (descending) or $\overleftarrow{A}_{n+1} = \overleftarrow{A}_n \backslash \overleftarrow{C}_{n+1}$ (ascending).
14. If $\vec{A}_{n+1} \neq \phi,\ \ or\ \ \overleftarrow{A}_{n+1} \neq \phi$ then $n = n+1$, and proceed to Step (2).
15. Otherwise, end of the distillation.

During the same distillations, when advancing from step $k$ to step $k+1$, the linguistic cutoff level $(s_{t_c},\alpha_c)^{(k)}$ is replaced by $(s_{t_c},\alpha_c)^{(k+1)} < (s_{t_c},\alpha_c)^{(k)}$ as follows ($D_k$ is the remaining set of alternatives to rank):

$$\begin{matrix} (s_{t_c},\alpha_c)^{(k+1)} \\ = \max\limits_{\substack{\{\bar{\sigma}(a_i,a_j)+z((s_{t_c},\alpha_c)^{(k)})<(s_{t_c},\alpha_c)^{(k)}\} \\ a_i,a_j \in D_k,}} \bar{\sigma}(a_i,a_j) \end{matrix} \tag{40}$$

where $z((s_{t_c},\alpha_c)^{(k)}) = \Delta_{S^P}\left(\frac{\alpha(\Delta_{S^P}^{-1}((s_{t_c},\alpha_c)^{(k)}))}{h_P} + \beta\right)$.

The analyst can fix one value for the distillation coefficients $\alpha$ and $\beta$ before the computations. The standard values recommended in the literature are $\alpha = -0.15$ and $\beta = 0.30$.

We obtain two complete preorders at the end of the distillation procedure. In each preorder, the alternatives are regrouped in a partition of

**Table 16.** Power, weakness, and qualification scores for Iteration 2 in distillation 3

| $ei_iOei_j$ | $ei_5$ | $ei_7$ |
|---|---|---|
| $p_{D_0}^{(s_c,\alpha_c)^{(0)}}$ **- power** | 1 | 0 |
| $f_D^{(s_c,\alpha_c)^{(0)}}$ **- weakness** | 0 | 1 |
| $q_{D_0}^{(s_c,\alpha_c)^{(0)}}$ **- qualification** | 1 | -1 |

equivalence classes, forming a ranking from the best to the worst alternatives.

Each class includes at least one alternative. A partial preorder of the alternatives is constructed utilizing the intersection of both preorders, which specifies the comparisons between alternatives and emphasizes the possible incomparabilities as follows:

- Alternative $a_i$ is preferred to the alternative $a_j$ if $a_i$ belongs to a class not worse than alternative $a_j$ in both preorders and a better class for at least one of the two preorders.
- Alternative $a_i$ is indifferent to alternative $a_j$ if $a_i$ and $a_j$ belong to the same class in the two preorders.
- Alternatives $a_i$ and $a_j$ are incomparable if $a_i$ belongs to a class better than $a_j$ in one preorder and worse in the other or vice *versa*.

To illustrate the proposed method, we present in the following section a step-by-step example of the linguistic ELECTRE III method for ranking a set of alternatives.

## 4 An Illustrative Example

We will use a case study from [25] to demonstrate the proposed approach. This case study is an Environmental Impact Significance Assessment problem in which heterogenous data (qualitative and quantitative judgments) obtained from a DM are used to determine the environmental impacts that a set of projects or industrial activities can have on a petrol station's usual operations.

This case study aims to evaluate seven ecological effects that can occur between the interactions of four industrial activities and four environmental factors in a petrol station. The evaluation seeks to rank the identified impacts from the most to the least significant. Each step of the linguistic extension of the ELECTRE III method is explained below.

Step 1. Formulation of the multicriteria ranking problem. Given a set of activities from a petrol station $A = \{a_1$: The operation of petrol pumps, $a_2$: the operation of the car wash, $a_3$: the transport of fuel and materials, and $a_4$: the filling of fuel tanks$\}$ and four possible environmental factors $F = \{f_1$: Daily sound comfort, $f_2$: hydrocarbons in the air, $f_3$: public health and civic safety, and $f_4$: energy infrastructures$\}$ a set of seven possible environmental impacts that are triggered from the interaction between $A$ and $F$, was identified $EI = \{(a_1,f_2),(a_1,f_3),(a_2,f_1),(a_2,f_4),(a_3,f_1),(a_3,f_4),(a_4,f_1)\}$.

For convenience, we define $EI = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_6, ei_7\}$. For the assessment of the elements in *EI*, a DM, which has specific knowledge is public health, expressed his preferences on *EI* using diverse expression domains: Numerical(N), Interval-valued(I), or Linguistic (L) over a set of 10 criteria defined in Table 1. Note that the preference direction for all criteria is to maximize.

The DM uses a linguistic domain with five linguistic terms denoted by $S^5$ to express his/her preferences. Each linguistic term set is symmetrically and uniformly distributed, and its syntax is defined in the following form:

$$S^5 = \begin{cases} S_0 : VeryLow(VL),\ S_1 : Low(L),\ S_2 : Mediu \\ S_3 : High(H), S_4 : VeryHigh(VH). \end{cases} \quad (41)$$

Step 2. Collecting the heterogeneous information: The DM assessed each criterion for each impact in EI using a heterogeneous framework. The expression domain used for each criterion was according to its nature; for criteria $g_1, g_2, g_6, g_7, g_8, g_9$, and $g_{10}$ were used the linguistic terms in $S^5$; for criteria $g_3$ and $g_5$ were used a scale based on real numbers; meanwhile, for criterion $g_4$ was used an interval scale. The assessment made by the DM is presented in Table 2.

Step 3. Fusion of the heterogeneous information: The chosen linguistic domain to fuse

the information is $S^5$. The integrated information given by the DM is shown in Table 3.

Step 4. Computing linguistic difference values between unified assessments: The linguistic difference value between a pair of 2-tuple linguistic values is stated in the linguistic comparison scale $S^C$ presented in Figure 3. Linguistic difference values are calculated by Eq. (23).

Step 5. Computing linguistic concordance values: Linguistic concordance index concerning a criterion $g_k$, $C_k(ei_i, ei_j)$.

Calculations to get individual linguistic concordance values. Initially, the linguistic preference scale $S^P$ is chosen. After that, for each criterion $g_k$, its linguistic concordance function is performed (Eq. (31)) and its indifference and preference threshold parameters are defined in 2-tuple linguistic values in $S^C$.

Each linguistic concordance value for each alternative $ei_i$, with regards to alternative $ei_j$, over a criterion $g_k$, is calculated using the linguistic outranking function (Eq. 31).

The calculation of the linguistic outranking value for each criterion is given in a linguistic preference scale $S^P = \{S_0^P, \dots, S_k^P\}$ with nine linguistic terms. The inter-criteria parameters of $g_k, k = 1,2, \dots,10$ are presented in Table 4, which are described in $S^C$.

**Example 3**. Calculation of $\bar{C}_1(ei_1, ei_6)$. According to the descriptive example, the computations of the linguistic concordance indices (Definition 3.2) can be made in the following form: From Table 3, $\bar{g}_1(ei_1) = (L, 0)$ and $\bar{g}_1(ei_6) = (VH, 0)$, then:

$$\begin{aligned} D_S(ei_1, ei_6)_1 &= D_S\left(\bar{g}_1(ei_1), \bar{g}_1(ei_6)\right)_1 \\ &= D_S\left((L,0),(VH,0)\right)_1 \\ &= \Delta_{S^C}\left(\frac{\left(\Delta_{S_{BLTS}}^{-1}(VH,0) - \Delta_{S_{BLTS}}^{-1}(L,0)\right)+8}{2x8}\times 8\right) \\ &= \Delta_{S^C}\left(\frac{(4-1)+8}{2x8}\times 8\right) \\ &= \Delta_{S^C}(5.5) \\ &= (I_5^C, 0.5). \end{aligned} \tag{42}$$

Based on Eq. (31), since:

$$\left(I_5^C, 0\right)q \le D_S(ei_1, ei_6)_1 = \left(I_5^C, 0.5\right) \ge (I_6^C, 0)p. \tag{43}$$

Then from interpolation, we calculate:

$$\bar{C}_1(ei_1, ei_6) = \left(\frac{(I_6^C,0)p - \frac{8}{2} - \left((I_5^C, 0.5) - \frac{8}{2}\right)}{(I_6^C,0)p - \frac{8}{2} - \left((I_5^C,0)q - \frac{8}{2}\right)}\right) \times 8 = (I_4^C, 0). \tag{44}$$

In this way, it is possible to get the linguistic concordance indices $\bar{C}_k(ei_i, ei_j)$ on a criterion $g_k$ for all pairs of alternatives $(ei_i, ei_j)$, and, finally, display the linguistic concordance matrices for each criterion.

For example, on the criterion $g_1$ the linguistic concordance matrix is expressed in Table 5. *The comprehensive linguistic concordance index* $\bar{C}(ei_i, ei_j)$.

The comprehensive linguistic concordance index $\bar{C}(ei_i, ei_j)$ is computed using the weight vector:

$$W = (0.36, 0.24, 0.08, 0.04, 0.04, 0.04, 0.04, 0.04, 0.08, 0.04)$$

for the family of criteria. The value of $\bar{C}(ei_1, ei_6)$ is computed as follows:

$$\begin{aligned} \bar{C}(ei_1, ei_6) = \Delta_{S^P}\Big(&0.36(\Delta_{S^P}^{-1}(\bar{C}_1(ei_1, ei_6)) \\ &+ 0.24(\Delta_{S^P}^{-1}(\bar{C}_2(ei_1, ei_6)) \\ &+ 0.08(\Delta_{S^P}^{-1}(\bar{C}_3(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_4(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_5(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_6(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_7(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_8(ei_1, ei_6)) \\ &+ 0.08(\Delta_{S^P}^{-1}(\bar{C}_9(ei_1, ei_6)) \\ &+ 0.04(\Delta_{S^P}^{-1}(\bar{C}_{10}(ei_1, ei_6))\Big) \\ = (S_5^P, 0.8048). \end{aligned} \tag{45}$$

Proceeding in the same way, for all pairs of environmental impacts $(ei_i, ei_j)$ representing the illustrative example, the comprehensive linguistic concordance matrix is obtained (Table 6).

Step 6. Computing linguistic discordance values $\bar{d}_k(ei_i, ei_j)$. The linguistic veto thresholds have been defined on criteria $g_1$ and $g_2$. These criteria can give a linguistic discordance index that is not null.

**Example 4**. Calculation of $\bar{d}_1(ei_1, ei_6)$. The computations of the linguistic discordance indices (Definition3.3) can be made as follows: from Table 3 $\bar{g}_1(ei_1) = (L, 0)$ and $\bar{g}_1(ei_6) = (VH, 0)$, from example 2 $D_S(ei_1, ei_6)_1 = (I_5^C, 0.5)$.

**Table 17**. Crisp outranking relation for iteration 1 in distillation 4

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_4$ | $ei_7$ |
|---|---|---|---|---|
| $ei_1$ | 0 | 0 | 0 | 0 |
| $ei_2$ | 0 | 0 | 0 | 0 |
| $ei_4$ | 1 | 1 | 0 | 0 |
| $ei_7$ | 1 | 0 | 1 | 0 |

**Table 18.** Power, weakness, and qualification scores for Iteration 1 in distillation 4

| $ei_i O ei_j$ | $ei_1$ | $ei_2$ | $ei_4$ | $ei_7$ |
|---|---|---|---|---|
| $p_{D_0}^{(s_{t_c},\alpha_c)^{(0)}}$ - power | 0 | 0 | 2 | 2 |
| $f_D^{(s_{t_c},\alpha_c)^{(0)}}$ - weakness | 2 | 1 | 1 | 0 |
| $q_{D_0}^{(s_{t_c},\alpha_c)^{(0)}}$ - qualification | -2 | -1 | 1 | 2 |

Since $D_S(ei_1, ei_6)_1 = (I_5^C, 0.5) \leq (I_6^C, 0)_P$ then from Eq. 42 $\bar{d}_1(ei_1, ei_6) = (S_0^P, 0)$.

In the same way, with this computation process, it is possible to obtain the linguistic discordance indices $\bar{d}_k(ei_i, ei_j)$ on the criterion $g_k$ for all pairs of environmental impacts $(ei_i, ei_j)$ and display the linguistic discordance matrices for each criterion where it is defined a linguistic veto threshold. For instance, on the criterion $g_1$, the computed discordance matrix is defined in Table 7.

Step 7. Computing the linguistic outranking relation. Based on the comprehensive linguistic concordance matrix and the partial linguistic discordance matrices, the value of $\bar{\sigma}(ei_1, ei_6)$ is computed as follows:

Since $\bar{K}(ei_1, ei_6) = 0$ and $\forall k, \bar{d}_k(ei_1, ei_6) < \bar{C}(ei_1, ei_6)$, then, from Eq. (21), $\bar{\sigma}(ei_1, ei_6) = \bar{C}(ei_1, ei_6) = (S_5^P, 0.8048)$.

For all pairs of alternatives representing the illustrative example, the linguistic credibility matrix or linguistic outranking matrix is obtained (see Table 8).

Step 8. Ranking of alternatives from the linguistic outranking relation $O_A^{\sigma}$. The ranking algorithm can be applied according to the linguistic credibility matrix obtained by the linguistic ELECTRE III (Table 6).

For illustration purposes, we describe the procedure followed to perform the first four descending distillations as follows:

Let:

$$\overrightarrow{EI_0} = EI = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_6, ei_7\}, z(\bar{\lambda}) = \Delta_{S^p}\left(\frac{\alpha\left(\Delta_{S^p}^{-1}(\bar{\lambda})\right)}{h_p} + \beta\right). \quad (46)$$

With $\alpha = -0.15, \beta = 0.30$.

Distillation 1.

Step 1: Let $n = 0, \overrightarrow{EI_0} = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_6, ei_7\}$, $k = 0$, then $(S_{t_c}, \alpha_c)^{(0)} = \max_{\substack{a,b \in \overrightarrow{EI_0} \\ a \neq b}} \bar{\sigma}(a,b) = (S_8^P, 0)$, and $D_0 = \overrightarrow{EI_0}$, hence $(S_{t_c}, \alpha_c)^{(1)} = \max_{\substack{\{\sigma(a,b)+z((S_{t_c},\alpha_c)^{(0)}) < (S_{t_c},\alpha_c)^{(0)}\} \\ a,b \in D_0}} \bar{\sigma}(a,b) = (S_7^P, 0.84)$. Given this linguistic cutoff, we can create a linguistic crisp outranking relation using Eq. (36). Table 9 shows the resulting crisp outranking relation. From the crisp outranking relation, we calculate the $(S_{t_c}, \alpha_c)^{(1)}$-qualifications $\left(q_A^{(S_{t_c},\alpha_c)^{(0)}}(a)\right) \forall a \in D_0$ Table 10 shows the calculated power, weakness, and qualification.

The maximum $(S_{t_c}, \alpha_c)^{(k+1)}$-qualification score is $\vec{q}_{D_k} = \max_{x \in D_k} q_{D_k}^{(S_{t_c},\alpha_c)^{(k)}}(x) = 4$, then: $\vec{D}_{k+1} = \left\{a \in D_k \middle| q_{D_k}^{(S_{t_c},\alpha_c)^{(k+1)}}(a) = \vec{q}_{D_k}\right\}$, $\vec{D}_1 = \{ei_6\}$. Because $|\vec{D}_1| = 1$ then $\vec{C}_1 = \vec{D}_1 = \{ei\}$ and the first distillation is completed. For the subsequent distillation, put $\overrightarrow{EI_1} = \overrightarrow{EI_0} \backslash \vec{C}_1 = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_6, ei_7\} \backslash \{ei_6\} = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_7\}$ and do do $n = n + 1 = 1$.

Distillation 2.

Step 1: Let $k = 0, D_0 = \overrightarrow{EI_1} = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_7\}$, then $(S_{t_c}, \alpha_c)^{(0)} = \max_{\substack{a,b \in \overrightarrow{EI_1} \\ a \neq b}} \bar{\sigma}(a,b) = (S_8^P, 0)$, and $(S_{t_c}, \alpha_c)^{(1)} = \max_{\substack{\{\sigma(a,b)+z((S_{t_c},\alpha_c)^{(0)}) < (S_{t_c},\alpha_c)^{(0)}\} \\ a,b \in D_0}} \bar{\sigma}(a,b) = (S_7^P, 0.84)$.

From this linguistic cutoff, we create the linguistic crisp outranking relation shown in Table 11. Then we calculate the $(S_{t_c}, \alpha_c)^{(1)}$-qualifications $\left(q_A^{(S_{t_c},\alpha_c)^{(0)}}(a)\right) \forall a \in D_0$ . Table 12 shows the calculated power, weakness, and qualification for $D_0$ in step

one of distillation 2. The maximum $(S_{t_c}, \alpha_c)^{(k+1)}$-qualification score is $\vec{q}_{D_k} = \max_{x \in D_k} q_{D_k}^{(S_{t_c}, \alpha_c)^{(k)}}(x) = 3$, then $\vec{D}_{k+1} = \left\{ a \in D_k \middle| q_{D_k}^{(S_{t_c}, \alpha_c)^{(k+1)}}(a) = \vec{q}_{D_k} \right\}$, $\vec{D}_1 = \{ei_3\}$.

Because $|\vec{D}_1| = 1$ then $\vec{C}_2 = \vec{D}_1 = \{ei_3\}$ and the first distillation is completed. For the subsequent distillation, put $\overrightarrow{EI}_2 = \overrightarrow{EI}_1 \backslash \vec{C}_2 = \{ei_1, ei_2, ei_3, ei_4, ei_5, ei_7\} \backslash \{ei_3\} = \{ei_1, ei_2, ei_4, ei_5, ei_7\}$ and do do $n = n + 1 = 2$.

Distillation 3.

Iteration 1: Let $k = 0, D_0 = \overrightarrow{EI_2} = \{ei_1, ei_2, ei_4, ei_5, ei_7\}$, then $(S_{t_c}, \alpha_c)^{(0)} = \max_{\substack{a,b \in \overrightarrow{EI_2} \\ a \neq b}} \bar{\sigma}(a,b) = (S_8^P, 0)$, and $(S_{t_c}, \alpha_c)^{(1)} = \max_{\{\sigma(a,b) + z((S_{t_c}, \alpha_c)^{(0)}) < (S_{t_c}, \alpha_c)^{(0)}\}} \bar{\sigma}(a,b) = (S_7^P, 0.84)$.

From this linguistic cutoff, we create the linguistic crisp outranking relation shown in Table 13. Then we calculate the $(S_{t_c}, \alpha_c)^{(1)}$-qualifications $\left( q_A^{(S_{t_c}, \alpha_c)^{(0)}}(a) \right) \forall a \in D_0$. Table 14 shows the calculated power, weakness, and qualification for $D_0$ in Iteration one of distillation 3. The maximum $(S_{t_c}, \alpha_c)^{(k+1)}$-qualification score is $\vec{q}_{D_k} = \max_{x \in D_k} q_{D_k}^{(S_{t_c}, \alpha_c)^{(k)}}(x) = 2$, then $\vec{D}_{k+1} = \left\{ a \in D_k \middle| q_{D_k}^{(S_{t_c}, \alpha_c)^{(k+1)}}(a) = \vec{q}_{D_k} \right\}$, $\vec{D}_1 = \{ei_5, ei_7\}$. Because $|\vec{D}_1| > 1$ we proceed with another iteration in distillation three.

Iteration 2: Let $k = 1$, and $(S_{t_c}, \alpha_c)^{(2)} = \max_{\{\sigma(a,b) + z((S_{t_c}, \alpha_c)^{(1)}) < (S_{t_c}, \alpha_c)^{(1)}\}} \bar{\sigma}(a,b) = (S_6^P, 0.24)$.

From this linguistic cutoff, we create the linguistic crisp outranking relation shown in Table 15. Then we calculate the $(S_{t_c}, \alpha_c)^{(2)}$-qualifications $\left( q_A^{(S_{t_c}, \alpha_c)^{(1)}}(a) \right) \forall a \in D_1$. Table 16 shows the calculated power, weakness, and qualification for $\vec{D}_1$ in Iteration 2 of distillation 3.

The maximum $(S_{t_c}, \alpha_c)^{(k+1)}$-qualification score is $\vec{q}_{D_k} = \max_{x \in D_k} q_{D_k}^{(S_{t_c}, \alpha_c)^{(k)}}(x) = 1$, then:

$$\vec{D}_{k+1} = \left\{ a \in D_k \middle| q_{D_k}^{(S_{t_c}, \alpha_c)^{(k+1)}}(a) = \vec{q}_{D_k} \right\}, \quad (47)$$

$$\vec{D}_1 = \{ei_5\}.$$

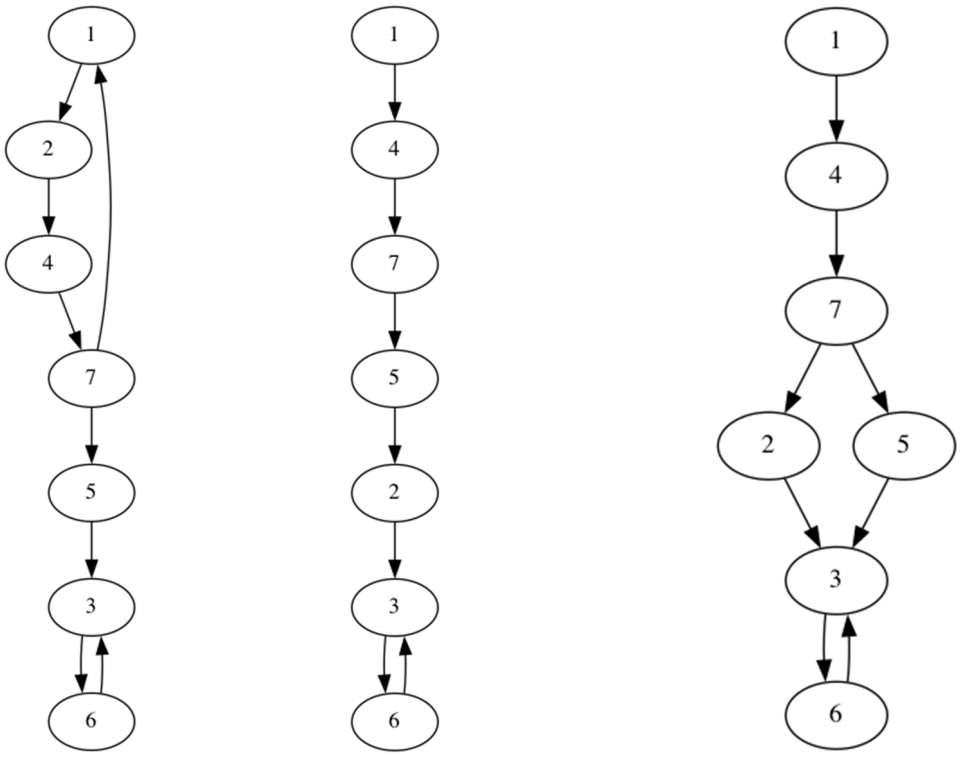
Because $|\vec{D}_1| = 1$ then $\vec{C}_3 = \vec{D}_1 = \{ei_5\}$ and the third distillation is completed. For the subsequent distillation, put $\overrightarrow{EI}_3 = \overrightarrow{EI}_2 \backslash \vec{C}_3 = \{ei_1, ei_2, ei_4, ei_5, ei_7\} \backslash \{ei_5\} = \{ei_1, ei_2, ei_4, ei_7\}$ and do $n = n + 1 = 3$.

Distillation 4

Iteration 1: Let $k = 0, D_0 = \overrightarrow{EI_3} = \{ei_1, ei_2, ei_4, ei_7\}$, then $(S_{t_c}, \alpha_c)^{(0)} = \max_{\substack{a,b \in \overrightarrow{EI_3} \\ a \neq b}} \bar{\sigma}(a,b) = (S_8^P, 0)$, and $(S_{t_c}, \alpha_c)^{(2)} = \max_{\{\sigma(a,b) + z((S_{t_c}, \alpha_c)^{(1)}) < (S_{t_c}, \alpha_c)^{(1)}\}} \bar{\sigma}(a,b) = (S_7^P, 0.84)$. From this linguistic cutoff, we create the linguistic crisp outranking relation shown in Table 17. Then, we calculate the $(S_{t_c}, \alpha_c)^{(1)}$-qualifications $\left( q_A^{(S_{t_c}, \alpha_c)^{(0)}}(a) \right) \forall a \in D_0$ . Table 18 shows the calculated power, weakness, and qualification for $\vec{D}_0$ in Iteration one of distillation 4. The maximum $(S_{t_c}, \alpha_c)^{(k+1)}$-qualification score is $\vec{q}_{D_k} = \max_{x \in D_k} q_{D_k}^{(S_{t_c}, \alpha_c)^{(k)}}(x) = 1$, then $\vec{D}_{k+1} = \left\{ a \in D_k \middle| q_{D_k}^{(S_{t_c}, \alpha_c)^{(k+1)}}(a) = \vec{q}_{D_k} \right\}$, $\vec{D}_1 = \{ei_7\}$. Because $|\vec{D}_1| = 1$ then $\vec{C}_4 = \vec{D}_1 = \{ei_7\}$ and the third distillation is completed. For the subsequent distillation, put $\overrightarrow{EI}_4 = \overrightarrow{EI}_3 \backslash \vec{C}_4 = \{ei_1, ei_2, ei_4, ei_7\} \backslash \{ei_7\} = \{ei_1, ei_2, ei_4\}$ and do do $n = n + 1 = 4$.

The remaining steps for the next descending distillations and the ascending distillation steps are processed in the same way. After completing the descending and ascending distillations, we got two complete preorders whose intersection creates the final ranking of the alternatives.

Figure 4 depicts the two preorders (descending and ascending distillations) calculated with the distillation procedure. In the descending preorder (Fig 4. a), there is an equivalence class in the first rank with the environmental impacts $ei_1, ei_2, ei_4, ei_7$, followed by EI $ei_5$ in the second rank, and at the last rank, there is an equivalence class with the environmental impacts $ei_3$ and $ei_6$. Meanwhile, the ascending distillation (Fig 4. b) is more granulated with $ei_1$ in the first rank, followed by $ei_4 \succ ei_7 \succ ei_5 \succ ei_2$, and at the last rank are $ei_3$ and $ei_6$.




a) Preorder resulting from a descending distillation

b) Preorder resulting from an ascending distillation

c) Final partial preorder

**Fig. 4.** Graphical representation of the preorders

Figure 4. c depicts the final preorder resulting from the intersection of the two preorders.

The final partial preorder follows a decreasing order of preferences, meaning that environmental impacts at the top are more significant than those at the bottom.

Hence, the final rank suggests that the $ei_1$ is the most significant environmental impact that is the interaction between action $a_1$ and factor $f_2$; in the second position is $ei_4$, that is the interaction between action $a_2$ and factor $f_4$; in the third position is $ei_7$ that is the interaction between action $a_4$ and factor $f_1$; in the fourth position are $ei_2$ and $ei_5$ that are the interactions between $a_1$ and factor $f_3$, and $a_3$ and factor $f_1$ respectively; it should be noted that although $ei_2$ and $ei_5$ are in the same ranking, they are incomparable. Thus, more analysis should be made for these two actions; finally, in the last rank, there is an equivalence class with $ei_3$ and $ei_6$ that suggests that both are indifferent to each other.

## 5 Conclusions

This paper aimed to develop a linguistic extension of the ELECTRE III method that allows solving instances of the multicriteria ranking problem with input data defined in heterogeneous contexts.

The new proposal fuses the heterogeneous information into 2-tuple linguistic values, allowing the DM to provide their preferences using diverse expression domains, such as numerical domain, interval-valued domain, and linguistic domain, according to the nature and uncertainty of the decision criteria, and their level of knowledge and experience.

Consequently, the new method is appropriate to integrate quantitative and qualitative criteria and uncertain information into the elements of the multicriteria model. In the modeling process of the ELECTRE III linguistic method, concordance, discordance, and credibility indices are proposed to consider linguistic inputs and outputs. Also, a linguistic difference function is stated to compute the linguistic difference between a pair of 2-tuple linguistic values. The output of the linguistic difference function is the input of the linguistic concordance and discordance indices.

Therefore, the linguistic extension of the ELECTRE III method offers good quality interpretability and understanding throughout the decision-making process in instances of the multicriteria ranking problem where there is heterogeneous data, as demonstrated in the illustrative example presented in this document.

The proposed methodology is applicable to real-life situations that involve decision-making with multiple conflicting criteria. This methodology can be used for various purposes such as project selection, supplier selection, job candidate evaluation, product design, environmental policy, and more.

When making decisions in contexts that involve diverse perspectives or input data from various sources, applying the linguistic ELECTRE III method can significantly impact decision-making in business, government, or social environments. For example, it can enhance the consideration of decision-maker preferences, improve transparency and accountability, facilitate cross-sector collaboration, and enable adaptation to dynamic environments.

The linguistic ELECTRE III can help organizations and policymakers navigate complex situations, manage uncertainty, and make more informed and equitable decisions in various business, government, and social environments. It provides a systematic and structured approach to decision-making in diverse contexts.

Soon, we plan to develop a linguistic extension of the ELECTRE III method for a collaborative group of DMs, and a hierarchical linguistic extension of the ELECTRE III method. Also, it is contemplated to carry out more real-world applications of the multicriteria ranking problem using our proposal.

## References

1. **Figueira, J. R., Greco, S., Roy, B., Słowiński, R. (2010).** Handbook of multicriteria analysis. Applied Optimization, Springer Berlin Heidelberg, Vol. 41, pp. 51–89. DOI: 10.1007/978-3-540-92828-7.
2. **Greco, S., Figueira, J., Ehrgott, M. (2016).** Multiple criteria decision analysis: state of the art surveys. International Series in Operations Research & Management Science, Springer New York, Vol. 37, pp. 155–185. DOI: 10.1007/978-1-4939-3094-4.
3. **Bouyssou, D., Vincke, P. (1997).** Ranking alternatives on the basis of preference relations: a progress report with special emphasis on outranking relations. Journal of Multi-Criteria Decision Analysis, Vol. 6, No. 2, pp. 77–85. DOI: 10.1002/(sici)1099-1360(199703)6:2<77::aid-mcda144>3.0.co;2-i.
4. **Espinilla, M., Halouani, N., Chabchoub, H. (2015).** Pure linguistic PROMETHEE I and II methods for heterogeneous MCGDM problems. International Journal of Computational Intelligence Systems, Vol. 8, No. 2, pp. 250. DOI: 10.1080/18756891.2015.1001949.
5. **Espinilla, M., Liu, J., Martínez, L. (2011).** An extended hierarchical linguistic model for decision-making problems. Computational Intelligence, Vol. 27, No. 3, pp. 489–512. DOI: 10.1111/j.1467-8640.2011.00385.x.
6. **Martinez, L., Herrera, F. (2000).** A 2-tuple fuzzy linguistic representation model for computing with words. IEEE Transactions on Fuzzy Systems, Vol. 8, No. 6, pp. 746–752. DOI: 10.1109/91.890332.
7. **Herrera, F., Martínez, L. (2000).** A 2-tuple fuzzy linguistic representation model for computing with words. Fuzzy Systems, Vol. 8. No. 6, pp. 746–752. DOI: 10.1109/91.890332.
8. **Herrera, F., Martinez, L. (2000).** An approach for combining linguistic and numerical information based on the 2-tuple fuzzy linguistic representation model in decision-making. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems,

Vol. 08, No. 05, pp. 539–562. DOI: 10.1142/s0218488500000381.

9. **Herrera, F., Martínez, L., Sánchez, P. (2005).** Managing non-homogeneous information in group decision making. European Journal of Operational Research, Vol. 166, No. 1, pp. 115–132. DOI: 10.1016/j.ejor.2003.11.031.
10. **Dutta, B., Guha, D., Mesiar, R. (2015).** A model based on linguistic 2-tuples for dealing with heterogeneous relationship among attributes in multi-expert decision making. IEEE Transactions on Fuzzy Systems, Vol. 23, No. 5, pp. 1817–1831. DOI: 10.1109/tfuzz.2014.2379291.
11. **Sohaib, O., Naderpour, M., Hussain, W., Martinez, L. (2019).** Cloud computing model selection for e-commerce enterprises using a new 2-tuple fuzzy linguistic decision-making method. Computers & Industrial Engineering, Vol. 132, pp. 47–58. DOI: 10.1016/j.cie.2019.04.020.
12. **Martinez, L., Herrera, F. (2012).** An overview on the 2-tuple linguistic model for computing with words in decision making: Extensions, applications and challenges. Information Sciences, Vol. 207, pp. 1–18. DOI: 10.1016/j.ins.2012.04.025.
13. **Leyva-López, J. C., Solano-Noriega, J. J., Figueira, J. R., Liu, J., Gastélum-Chavira, D. A. (2021).** Non-dominated sorting genetic-based algorithm for exploiting a large-sized fuzzy outranking relation. European Journal of Operational Research, Vol. 293, No. 2, pp. 615–631. DOI: 10.1016/j.ejor.2020.12.026.
14. **Roy, B. (1991).** The outranking approach and the foundations of electre methods. Theory and Decision, Vol. 31, No. 1, pp. 49–73. DOI: 10.1007/bf00134132.
15. **Rivera, G., Florencia, R., Guerrero, M., Porras, R., Sánchez-Solís, J. P. (2021).** Online multi-criteria portfolio analysis through compromise programming models built on the underlying principles of fuzzy outranking. Information Sciences, Vol. 580, pp. 734–755. DOI: 10.1016/j.ins.2021. 08.087.
16. **Rivera, G., Porras, R., Sanchez-Solis, J. P., Florencia, R., García, V. (2022).** Outranking-based multi-objective PSO for scheduling unrelated parallel machines with a freight industry-oriented application. Engineering Applications of Artificial Intelligence, Vol. 108, pp. 104556. DOI: 10.1016/j.engappai.2021.104556.
17. **Roy, B. (1996).** Multicriteria methodology for decision aiding. Nonconvex Optimization and Its Applications, Springer US, Vol. 12, pp. 277–293. DOI: 10.1007/978-1-4757-2500-1.
18. **Martinez, L., Liu, J., Yang, J. (2006).** A fuzzy model for design evaluation based on multiple criteria analysis in engineering systems. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol. 14, No. 03, pp. 317–336. DOI: 10.1142/s0218488506004035.
19. **Martínez, L., Rodríguez, R. M., Herrera, F. (2015).** 2-tuple linguistic model. pp. 23–42. DOI: 10.1007/978-3-319-24714-4_2.
20. **Merigó, J. M., Gil-Lafuente, A. M. (2013).** Induced 2-tuple linguistic generalized aggregation operators and their application in decision-making. Information Sciences, Vol. 236, pp. 1–16. DOI: 10.1016/j.ins.2013.02.039.
21. **Wei, G. W. (2011).** Some harmonic aggregation operators with 2-tuple linguistic assessment information and their application to multiple attribute group decision making. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol. 19, No. 06, pp. 977–998. DOI: 10.1142/s0218488511007428.
22. **Dubois, D., Prade, H. (2008).** An introduction to bipolar representations of information and preference. International Journal of Intelligent Systems, Vol. 23, No. 8, pp. 866–877. DOI: 10.1002/int.20297.
23. **Roy, B. (1978).** Un algorithme de classement fondé sur une représentation floue des préférences en présence de critères multiples. Cahiers du CERO, Vol. 20, No. 1, pp. 3–24. DOI: 10.1007/0-387-23081-5_4.
24. **Zulueta, Y., Rodríguez, D., Bello, R., Martínez, L. (2016).** A linguistic fusion approach for heterogeneous environmental impact significance assessment. Applied

Mathematical Modelling, Vol. 40, No. 2, pp. 1402–1417. DOI: 10.1016/j.apm.2015.07.016.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is Juan Carlos Leyva López.*

ISSN 2007-9737

# Architecture of a Socio-Emotional Conversational Agent Capable of Identifying Intentions within a Socratic Dialogue

Héctor Rodríguez-Arteaga, Bárbara María Esther García-Morales,
María Lucila Morales-Rodríguez*, Claudia Guadalupe Gómez-Santillán

Tecnológico Nacional de México,
Instituto Tecnológico de Ciudad Madero,
México

{G16070821, D10070255, lucila.mr, claudia.gs}@cdmadero.tecnm.mx

**Abstract.** This article presents an architecture for developing a socio-emotional conversational agent to identify intentions within Socratic dialogues, using a combination of speech act theory and sentiment analysis. This proposal seeks to create a socio-emotional conversational agent capable of addressing the issue of how emotions associated with personality affect the recognition of intentions of expressions within the educational environment. To identify the appropriate intention, this approach considers the influence of personality, the polarity of the words used by the user, and the context in which the dialogue is situated. With this, it is expected to accurately identify communicative intentions, facilitating a natural and empathetic interaction between intelligent virtual agents and users, contributing to the field of artificial intelligence applied to education.

**Keywords.** Social-emotional conversational agents, speech act theory, sentiment analysis, Socratic dialogue, artificial intelligence.

## 1 Introduction

Nowadays, intelligent agents are used in various applications, such as virtual assistants (Siri, Alexa, Google Assistant), customer service chatbots, and personalized online tutors. These agents not only facilitate daily tasks but also promise to transform our interaction with technology radically. The integration of advanced concepts like Speech Act Theory (SAT) and Sentiment Analysis (SA) in these agents points towards an improvement in the ability to discern intentions and emotions, marking an advancement in creating dialogues that allow for more natural and empathetic interaction. According to [1, 5], Speech Act Theory (SAT) enables the processing and generation of language to reflect specific communicative intentions, which can be applied in the dialogue management models of conversational agents.

On the other hand, studies such as those by [4, 11] highlight the relevance of SA for capturing polarity and emotions in text, which would improve their understanding of the specific context established in a conversation. No research has been found in the literature review that explores how emotions specifically affect the determination of intentions in expressions within virtual educational environments.

This presents an opportunity to understand the intentions in complex conversational interactions that characterize human communication. The work of [3, 5] highlights the importance of integrating speech act theory into the design of agents. However, a deep exploration of how this approach can work with sentiment analysis in identifying intentions in educational contexts is still lacking. To address this issue, this article proposes an architecture for a socio-emotional conversational agent with the role of an educational coach.

This architecture combines speech act theory (SAT) and sentiment analysis (SA) to identify intentions under the influence of personality profiles and polarity in expressions within the context of educational coaching. This article presents several vital sections reviewing related works on socio-emotional conversational agents to establish the context and identify the need for architecture development. It then provides the background, where the fundamental concepts related to the techniques above are detailed.

It then describes how the architecture of the Agent with the role of virtual coach is composed and the functions of each module. Finally, the conclusions about the functionality of the Agent, its scope, and how it contributes to the advancement of socio-emotional conversational agents in educational environments are discussed.

## 2 Related Works

This section presents some related works in the area of socio-emotional conversational agent development that use speech act theory as a strategy to characterize phrases in intelligent virtual agents. In [9], relevance theory is adopted as the strategy for analyzing phrases, focusing on how human cognition and the relevance of the context assist in understanding communicative intention and its interpretation. Speech acts represent communicative intention and guide the actions of an intelligent agent in a dialogue.

The dialogue is organized into "communication games" and "phases"; these communication games help organize the speech acts used by speakers during the interaction, characterizing the phrases through perlocutionary acts that represent goals, illocutionary acts that define the strategy used in the game, and the conditions that can change it. In [3, 7], the use of SAT (Speech Act Theory) in selecting phrases from a specific corpus for virtual agents with personality is highlighted.

In [3], a manually characterized corpus of phrases was analyzed, including aspects of speech acts, emotions, personality, and other relevant attributes, using the MBTI model to simulate the personality of the virtual Agent. This allowed for fine-tuning in phrase selection, reflecting an accurate alignment with the Agent's personality traits.

In [7], previous work is improved by incorporating an automated phrase selection model for a virtual coach, which stands out for combining SAT with an inductive educational methodology. This model generates powerful questions inspired by Socratic dialectics, adapted to the coachee's personality through the MPBCD model (Personality Model Under a Decision Context). Despite the significant contributions of the mentioned works in applying Speech Act Theory (SAT) and other theoretical frameworks for developing intelligent virtual agents, there is an issue in identifying intentions, as these studies assume that the intention behind the user's expressions is known.

However, no detailed mechanism for identifying them is available. This affects the agents' ability to understand and respond appropriately to the complexity and subtlety of dialogue since effective interaction crucially depends on recognizing not only what is explicitly said but also hidden intentions and communicative nuances.

Without understanding these subtle aspects, agents may respond inappropriately, making the conversation less natural and effective. This work incorporates a mechanism for identifying communicative intentions through sentiment analysis and the theory of speech acts.

This mechanism is situated within the architecture of a socioemotional conversational agent, which can identify the intentions in contrast to the user's personality profile and the context in which the conversation takes place, obtaining a deep and complete understanding of the context. With the integration of this mechanism, we intend to overcome the observed weaknesses by significantly improving the interaction between the conversational Agent and the user, allowing a more dynamic and empathic dialogue.

## 3 Background

To thoroughly understand the techniques that make up the virtual tutor capable of identifying intentions within a Socratic dialogue, it is essential to detail the fundamental concepts related to conversational agents, the theory of speech acts, the influence of personality models on socioemotional agents, Socratic dialectics in educational coaching processes and sentiment analysis.

### 3.1 Conversational Agents as Virtual Coaches

Conversational agents, also known as chatbots or virtual assistants, are artificial entities capable of emulating human behavior. As described by [5], these agents must be designed to interpret the intentions of conversations and understand the

interlocutors' questions and answers to decide what actions to take in real time. Natural language processing techniques, as well as machine learning methods and algorithms, are used to deal with that [9].

These agents incorporate the fundamental principles underlying most intelligent systems in their architecture. These principles include reactivity, the management of an internal state, the formulation of principles and goals, the capability for autonomy in decision-making, a predisposition towards sociability, and the ability of reasoning. These principles link the agents to the real world through beliefs, desires, and intentions.

The Agent analyzes its personality and context to define its identity and act according to the context in which it operates. The desires are closely connected with the goals that the Agent pursues. In this application case, they focus on formulating questions based on dialectics influenced by the Agent's personality [7].

### 3.2 Speech Act Theory for Phrase Characterization in Dialogue

Currently, conversational agents that employ speech act analysis and computational linguistics are available to deduce the meaning of words contained in expressions provided by the user [5].

Proposed by [1, 13], Speech Act Theory (SAT) is a branch of linguistic pragmatics focused on analyzing how people use language to perform actions in the world beyond merely communicating information. A mechanism to recognize the intention of phrases the user expresses is characterizing attributes belonging to speech acts [3, 6, 9].

According to [1], phrases can be characterized into three main types of acts (locutionary, illocutionary, and perlocutionary). Illocutionary speech acts play a fundamental role in determining the underlying meaning in a sentence, as pointed out by [13], illocutionary acts are divided into two categories: direct and indirect, and the main difference lies in that, in indirect acts, the locutionary and illocutionary acts do not overlap, while in direct acts, they do.

Furthermore, it's important to highlight that these illocutionary acts can be classified into five distinct categories: assertive, commissive, directives, declaratives, and expressive. According to [13], illocutionary acts are actions carried out by speakers when expressing words with communicative force. These acts allow the characterization of a sentence and identify the speaker's intention, which influences the receiver's interpretation.

Speech acts can characterize any type of dialogue, including a Socratic dialogue. In this type of dialogue, the phrases are direct acts, implying that the coachee's intention coincides with the coach's understanding. The illocutionary verbs in this type of dialogue can play locutionary or perlocutionary roles.

The powerful questions in this type of dialogue are classified as directive illocutionary acts aimed at interrogating [5]. This type of interaction in conversational agents can be linked with traits and personality types using personality models under decision contexts such as the MPBCD [7].

### 3.3 Personality Models for Virtual Coaches

To emulate a reflection of the coachee's personality within the educational coaching processes, it is necessary to resort to behavior modeling [7] to achieve effective realism. Therefore, it is essential to integrate elements such as emotions, personality traits, preferences, and motivations into the personality modeling functions of the virtual entity [12].

To achieve this, one can make use of the two major theories of personality on human behavior which are based on traits (FFM, OCEAN) and based on types (MBTI, KTS). These theories are grounded in psychological research dedicated to characterizing and describing individuals' personalities [2].

According to personality theory, the FMM and OCEAN models are based on personality traits. They are highlighted through five factors (openness, conscientiousness, extraversion, agreeableness, and neuroticism) representing personality [12]. These factors are derived from an analysis of questionnaires and adjectives that indicate the strengths and weaknesses of each trait through levels (high-low) [2].

The MBTI model is based on personality types, assigning individuals with abbreviated four-letter labels, thereby creating 16 different labels or

ISSN 2007-9737



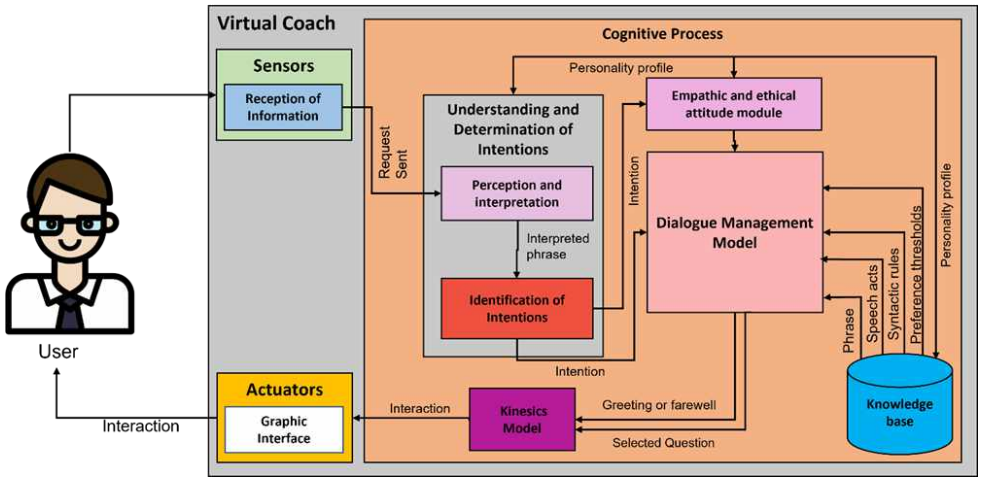



**Fig. 1.** Architecture of the conversational agent with the role of virtual coach

profiles. The labels are generated based on information obtained through a questionnaire. Each label describes the attitude of individuals in their interaction with their environment, which is manifested through dichotomies [2].

Another model based on personality types is the KTS model. That is a self-assessed questionnaire designed to help people understand their personality. KTS is based on MBTI, using the same combinations of items to establish the labels. The difference between this model and MBTI is the inclusion of temperament, role, and role variants [2].

The MPBCD (Personality Model Under a Decision Context) model proposes a relationship between the FFM, MBTI, and KTS models to integrate the traits and personality types approaches. Its primary function is to use information from individuals and generate a profile that illustrates their behavior in decision-making situations while examining the influence of personality on their preferences [2]. The MPBCD has been employed to develop agents based on Socratic dialectics [5].

### 3.4 Socratic Dialectics in Educational Coaching Processes

The Socratic dialectic, known as maieutic, involves conducting a strategic interrogation and counterpointing each answer with additional questions. Often, these questions are formulated neutrally to reach a truthful answer that synthesizes the partial truths of the previous responses [10].

In [6], the Socratic dialectic characterizes a dialogue that incorporates powerful questions and relies on two essential components: maieutics and irony. Its purpose is to foster self-knowledge through the cognitive opportunities that emerge from the responses generated in the context of Socratic maieutics.

According to the literature, the powerful questions posed in this context can be linked both to speech acts and to existing personality profiles, allowing the emulation of the coachee's personality in each interaction.

### 3.5 Sentiment Analysis (SA)

Personality-related characteristics can influence decision-making when generating new phrases. To analyze the emotions conveyed in each sentence spoken by the coachee, it is proposed to work with the relationship between the degree of polarity and the intentions through the technique of Sentiment Analysis (SA).

Sentiment Analysis (SA), known as Opinion Extraction, Opinion Mining, Sentiment Mining, or Subjective Analysis, is a computational technique for studying opinions, sentiments, and emotions expressed in texts. SA covers three processing

tasks: the binary classification of the attitude of a text, the self-assessment of a text according to its degree of polarity, and the identification of the aspects mentioned in a text, together with their associated sentiments [11].

The binary classification of attitude in a text is a natural language processing technique used to discern whether a text expresses a positive or negative attitude toward a particular topic [4]. The self-assessment of a text-based on polarity degrees involves classifying messages as positive, negative, or neutral.

This representation is the most basic and widely used in sentiment analysis [8]. The authors [4] comment that this task requires advanced natural language processing techniques, such as syntactic dependency analysis and the extraction of named entities.

These techniques are combined with sentiment classification algorithms specifically adapted for each identified aspect. Considering the tasks of SA, it is possible to use it in a Socratic dialogue to analyze the emotions associated with the intentions expressed through speech acts deeply.

## 4 Proposed Architecture for the Creation of the Conversational Agent

A detailed architecture has been conceptualized and designed to initiate the development of the conversational Agent capable of identifying intentions in a Socratic dialogue, as shown in Figure 1. This architecture, grounded in the BDI (Belief, Desire, Intention) agent model, is based on the architecture proposed by [5], which compiles all the essential elements for the construction of the virtual tutor.

This system consists of several interconnected modules designed to interpret and respond to user interactions in a virtual educational coaching environment. The interaction process starts with the Information Reception Module, which captures the user's phrase, identifying if the type of illocutionary act is expressive.

The response is generated from the knowledge base (phrases for greetings or farewells). In other cases, if this class is not expressive, it sends the phrase to the Understanding and Intention Determination Model, which contains the Perception and Interpretation Module. This module receives and detects the polarity of the phrase from a neutral perspective, considering the influence of the personality profile in the interpretation.

This allows for a more precise definition of the intensity of that polarity from the perspective of the Agent's personality, which becomes the factor for determining the illocutionary force of the phrase, resulting in the identification of intentions.

If more than one intention is detected, they are passed to the Intention Inference Module, which uses the ID3 (Iterative Dichotomiser 3) algorithm to choose an intention according to the context of the conversation, taking into account the value of the polarity associated with the description of the personality profile.

This obtained intention is sent to the speech acts module, thus continuing with the determination of a powerful question that will serve as a response to the coachee.

The Knowledge Base stores greeting phrases, grammatical rules, the description of personality profiles, and information criteria that guide the generation of responses. The dialogue management model comprises the greeting or farewell selection process and the Dialectic Modeling, which includes modules for analyzing speech acts and selecting questions that promote reflection, taking into account the personality and preferences of the virtual Agent.

The Phrase Characterization Process through Speech Acts evaluates the issue raised by the user and the coachee's personality to generate powerful questions that reflect the coachee's profile. The Question Selector analyzes the issue at a grammatical level to structure these questions, which are filtered within the Dialogue Management Module through the Deliberative Process. The questions are selected according to the preferences of the virtual tutor, using the ELECTRE III method to determine the most suitable solution.

## 5 Discussion and Future Work

In this article, an architecture for the development of a socio-emotional conversational agent that integrates Speech Act Theory (SAT) and

Sentiment Analysis (SA) was proposed through the identification of the polarity of phrases provided by the user, associating it with the personality profile and the characterization of illocutionary forces to precisely identify intentions in a Socratic dialogue, especially in the context of educational coaching.

Combining these two techniques provides a richer understanding of the interlocutor's expressions, allowing the virtual tutor to make better decisions in its interaction, enriching the interaction and feedback in the educational coaching process.

As a virtual tutor, this conversational Agent can significantly contribute to developing conversational agents and credible virtual entities in educational environments. This advancement not only has applications in educational coaching but can also extend to other areas of education and human-machine interaction.

This represents an important step towards creating more intelligent and empathetic virtual entities that can understand and adapt to their interlocutors' emotional and cognitive needs. In order to assess the performance of the socio-emotional conversational Agent, we will conduct quantitative and qualitative analyses.

In the quantitative approach, we will compare the intentions perceived by real people with different personality profiles to the results generated by the Agent that simulates these same profiles. Additionally, we will evaluate the coherence of the Agent's responses by using a dictionary to compare the definitions with the responses obtained.

The qualitative approach will include semi-structured interviews with real people. That allows us to explore how they perceive and justify identifying intentions and how this relates to their personality profile.

## Acknowledgments

The authors want to thank Laboratorio Nacional de Tecnologías de Información and the support of (a) Cátedras CONAHCYT Program Number 3058, (b) the TecNM/Instituto Tecnológico de Ciudad Madero, and (c) CONAHCYT Scholarship for Postgraduate Studies with CVU 1239275.

## References

1. **Austin, J. L. (1975).** How to do things with words: second edition. Harvard University Press. books.google.com/books?id=u67VAAAAMAAJ.
2. **Castro-Rivera, J. A. (2017).** Modelado de personalidad en modelos preferenciales multicriterio a través de agentes virtuales inteligentes. Master's Thesis, Instituto Tecnológico de Ciudad Madero.
3. **Delgado-Hernández, X. (2021).** Aplicación de modelos de programación matemática a la selección de expresiones verbales en agentes virtuales socio-emocionales. Master's Thesis, Instituto Tecnológico de Ciudad Madero.
4. **Dubiau, L., Ale, J. M. (2013).** Análisis de sentimientos sobre un corpus en español: experimentación con un caso de estudio. Proceedings of the XIV Argentine Symposium on Artificial Intelligence, pp. 36–47.
5. **Garcia-Morales, B. M. E. (2021).** Modelo de selección de frases para un coach virtual que utiliza la técnica de la dialéctica. Master's Thesis, Instituto Tecnológico de Ciudad Madero.
6. **García-Morales, B. M. E, Morales-Rodríguez, M. L., Rangel-Valdez, N., García-Vite, P. M., González-Barbosa, J. J. (2022).** Caracterización de frases para un sistema conversacional inteligente en un entorno educativo virtual basado en las características de la dialéctica y los actos del habla. Research in Computing Science, Vol. 151, No. 6, pp. 105–118.
7. **García-Morales, B. M. E, Morales-Rodríguez, M. L., Valdez, N. R. (2023).** Conversational system based on Socratic dialectics in a virtual educational environment. Research in Computing Science, Vol. 152, No. 4, pp. 17–26.
8. **Gonzalvo, S. H. (2020).** Introducción a las técnicas de análisis de sentimientos. Bachelor's Thesis, Universidad de Granada
9. **Morales-Rodriguez, M. L. (2007).** Modèle d'interaction sociale pour des agents conversationnels animés. Application à la rééducation de patients cérébro-lésés

(Doctoral Thesis). Université Toulouse III: Paul Sabatier.

10. **Castillero-Mimenza, O. (2018).** Método Socrático: qué es y cómo se aplica en la psicología. Psicología clínica, Vol. 21, No. 43.
11. **Pang, B., Lee, L. (2008).** Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, Vol. 2, No. 1–2, pp. 1–135. DOI: 10.1561/1500000011.
12. **Pérez-Pinillos, D., Fernández, S., Borrajo, D. (2013).** Modeling motivations, personality traits and emotional states in deliberative agents based on automated planning. Communications in Computer and Information Science, Vol 271. DOI: 10.1007/978-3-642-29966-7_10.
13. **Searle, J. R. (1969).** Speech acts: An essay in the philosophy of language. Cambridge University Press. DOI: 10.1017/CBO9781139173438.

*Article received on 29/02/2024, accepted on 15/05/2024.*
**Corresponding author is Maria Lucila Morales Rodriguez.*

ISSN 2007-9737

# A Linear Genetic Programming Approach for the Internet Shopping Optimization Problem with Multiple Item Units (ISHOP-U)

Jazmin Del-Angel, Alejandro Santiago, Salvador Ibarra-Martínez, José Antonio Castán-Rocha*, Mayra Guadalupe Treviño-Berrones

Universidad Autónoma de Tamaulipas,
Facultad de Ingeniería Tampico,
Mexico

a2233338031@alumnos.uat.edu.mx, aurelio.santiago@uat.edu.mx, {sibarram,jcastan, mgtrevino}@docentes.uat.edu.mx

**Abstract.** Evolutionary computation (EC) is a broad field of artificial intelligence where evolutionary processes inspire algorithms, such as artificial immune systems, inspired by the evolution of acquired immune systems. The predominant approach in EC is Evolutionary Algorithms (EAs), inspired by the evolution of Darwin's natural species. A different approach is Evolutionary Programming (EP), which, instead of evolving individuals representing the problem decision variables (chromosomes), evolves programs, which code instructions, and executing those instructions generates a solution. Genetic Programming (GP) is an approach analogous to Genetic Algorithms (GAs), but it differs in that it works over programming instructions instead of decision variables. Although GP is an exciting approach, it is more complicated to implement due to the necessity of managing tree data structures. Linear Genetic Programming (LGP) is more straightforward than traditional GP, without the need for tree data structures. This chapter shows a proof of concept to implement LGP to evolve programs for the Internet Shopping Optimization Problem with multiple item Units (ISHOP-U), an NP-Hard optimization problem. Readers can easily implement the proposed approach and produce Linear Genetic Programming algorithms for other problems.

**Keywords.** ISHOP-U, evolutionary programming, linear genetic programming.

## 1 Introduction

Genetic Algorithms are the more expansive Evolutionary Algorithm (EA) approach, generally used for search and parameter optimization problems based on sexual reproduction and the principle of survival of the fittest. To solve a problem, we start from an initial set of individuals, called a population, generated randomly. Each of these individuals represents a possible solution to the problem.

These individuals will evolve through environmental selection and adapt their characteristics according to a fitness function, improving it after generations. The population evolves towards similar characteristics to achieve a particular goal or objective function. EAs have proved to be effective in addressing real-world complex optimization problems (e.g., [23, 24]).

The Artificial Immune Systems [7, 28, 30, 8, 27, 31, 11, 9, 1, 12, 19] is an example of an Evolutionary Computation algorithm not inspired by genetics evolution, inspired by the adaptation of biological immune systems [18]. The EA analogy is that individuals represent decision variables.

Another subfield of Evolutionary Computation is Evolutionary Programming [10] (EP), where the

**Algorithm 1** Genetic algorithms standard framework

**while** ¬ Stop condition **do**
  $Parent \leftarrow Selection(Pop)$
  $offSpring \leftarrow Crossover(Parent)$
  $Mutation(offSpring)$
  $Pop \leftarrow Pop \cup offSpring$
  $EnviromentalSelection(Pop)$
**end while**

**Algorithm 2** Matrix $S$ translation to program instructions

$Program \leftarrow$ an empty list of instructions
**for** $i = 1$ to $m$ **do**
  **for** $j = 1$ to $n$ **do**
    **if** $S_{i,j} > 0$ **then**
      $Program \leftarrow$ add the instruction $s_{i,j}$ equals to $S_{i,j}$ matrix value
    **end if**
  **end for**
**end for**
**return** $Program$

term programs replace individuals, and a program means a set of instructions to be executed, as in a computer program. In other words, EAs explicitly evolve the problem decision variables, while EP evolves operations that would implicitly conduct to the decision variables.

A subfield inside EP is Genetic Programming [4, 22, 13, 14, 15, 16] (GP). GP shares the sexual reproduction inspiration with the Genetic Algorithms (GAs). Therefore, the evolutionary operators of selection [3], crossover [26, 25, 20, 2], and mutation [29] also exist in GP, although a distinctive characteristic of GP is that chromosomes are of variable length. In addition, GP has the clone operator.

A clone operator is necessary because the environmental selection (survival of the fitness) is regularly different than in GAs, as the flowchart in Figure 1 shows, not allowing overlap in the generations between offspring and parents. Since GP inspiration is to evolve instructions and not decision variables, we found the above to be the authentic difference between the GA and GP approaches and not their frameworks. Although Genetic Programming [17] entails greater implementation complexity due to the management of tree data structures. Using tree data structures in GP requires tree traversal techniques to evaluate the programs, and the trees produced could be unbalanced, adding complexity.

Linear Genetic Programming [5] (LGP) offers a simpler alternative to traditional GP, dispensing with tree data structures; LGP evaluates programs sequentially. LGP uses variable-length linear data structures as the linked list to represent the programs, executing instructions sequentially in the list order, closing the gap between GP and the regular computer programming languages.

The assembly language has served as the metaphor for LGP; programs have registers (variables), read-only registers (constants), and instructions consist of operators (e.g., $+,-,\times,\div$) and operands, e.g., $r_3 = r_1 \times c_2$ where the destination register $r_3$ stores the result of the multiplication of the register $r_1$ with the constant register $c_2$.

We consider the GA framework well endorsed by artificial intelligence researchers and tested; we don't see a restriction not to use it directly with GP programs, as in Algorithm 1, instead of the non-overlapping between parents and offspring approach from the orthodox GP framework in Figure 1. Therefore, this work follows the GA framework instead of the orthodox GP flowchart. Generally speaking, in the words of John R. Koza: "The best computer program that appeared in any generation, the best-so-far solution, is designated as the result of genetic programming" [13].

This chapter presents a proof of concept for implementing LGP and developing programs that address the Internet Shopping Optimization Problem with Multiple Item Units [21] (ISHOP-U), an NP-Hard optimization problem. The proposed approach uses the GA framework for GP. It maps LGP instructions to the classical decision's variables representation in the GAs and the decision's variables to LGP instructions. The remainder of the chapter sections are as follows: Section 2 our Linear Genetic Programming proposal for the ISHOP-U.

ISSN 2007-9737



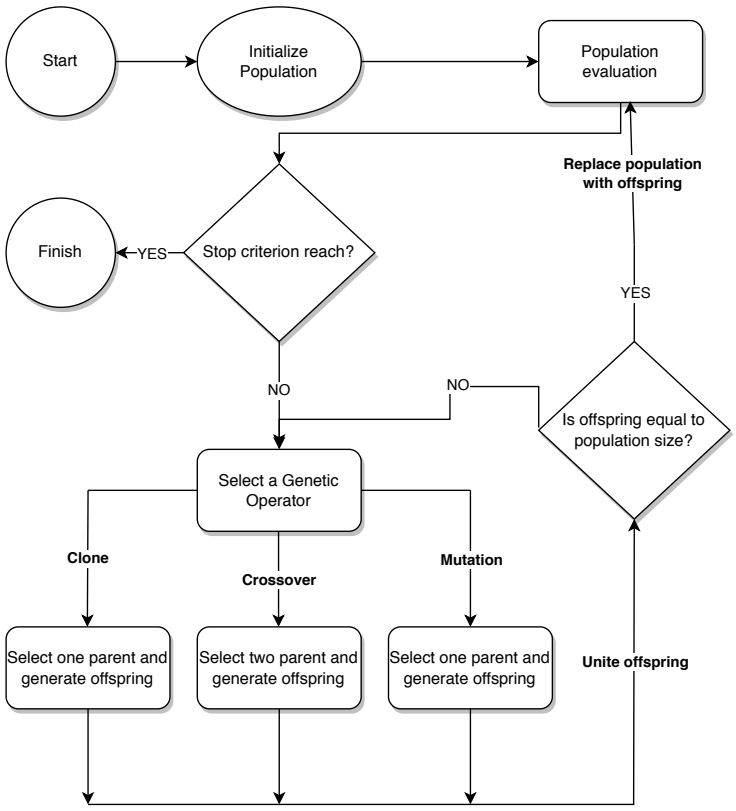



**Fig. 1.** Genetic programming orthodox flowchart

Section 3 is the experimental setup. Finally, Section 4 discusses and concludes the experimentation results.

## 2 Linear Genetic Programming for the ISHOP-U

This section describes our Linear Genetic Programming (LGP) proposal for the Internet Shopping Optimization Problem with multiple item Units (ISHOP-U). As the first attempt to apply LGP to the ISHOP-U, our approach is as simple as possible. Recalling the ISHOP-U formulation in [21] for a problem instance of $m$ stores and $n$ products, a candidate solution for the problem is a matrix $S$ of size $m \times n$; we consider their components $s_{i,j}$ as variable registers in the LGP.

As constant registers, we consider the components $a_{i,j}$ of the product availability matrix $A$ of size $m \times n$. The variable registers are all conditioned to be $s_{i,j} < a_{i,j}$, and the only operator implemented is assignation ($=$), e.g., $s_{i,j} = 5$, for a purchase of 5 units of the product $j$ in the store $i$. The description of the implementation guidelines and evolutionary operators appears below.

### 2.1 Population Initialization

The first population of programs has a random initial length (number of instructions) between $[0, m \times n]$. Every instruction is filled with an assignation to random register $s_{i,j}$ with a random integer between $[0, a_{i,j}]$, e.g., $s_{1,2} = 5$ if and only if $a_{1,2} \geq 5$.

**Table 1.** Parameter setting for the LGP implementation

| | **LGP** |
|---|---|
| Population size: | $100$ |
| MaxEvaluations: | $25,000$ |
| Selection: | Binary Tournament |
| Recombination: | $p_r = 1.0$ |
| Mutation: | $p_m = 0.05$ |

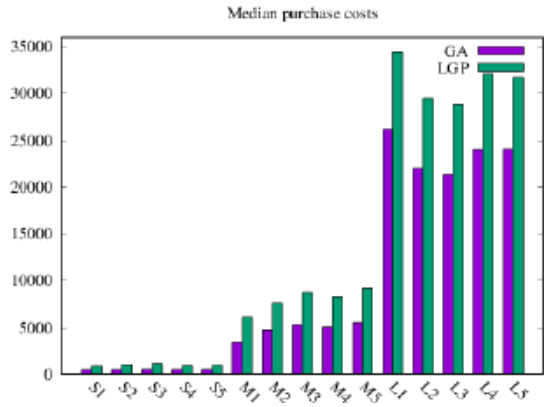



**Fig. 2.** Graph for the experimental results for the GA and LGP over the considered 15 instances

### 2.2 Crossover

Given two programs represented as linear vectors $\vec{x}$ and $\vec{y}$. The crossover operator copies in a single third vector $\vec{z}$ the individual instructions from $\vec{x}$ and $\vec{y}$. Later, the two new programs (child) inherit the instructions from $\vec{z}$. Every $\vec{z}$ instruction goes exclusively to the first or second child, with a uniform probability.

### 2.3 Mutation

The mutation operator uses a probability of mutation $p_m$ to change every program instruction. Once an instruction is decided to be changed, their register $s_{i,j}$ is reassigned to a random integer value between $[0, a_{i,j}]$.

### 2.4 Repair

This LGP proposal for the ISHOP-U uses the same repair method for unfeasible solutions as the Genetic Algorithm from [21].

The above is possible by executing the program instructions to load the matrix $S$ as an intermediate representation and then repairing $S$. Once repaired, $S$ is translated to program instructions using the following Algorithm 2.

Algorithm 2 has the advantage of producing a compact program representation in $O(m \cdot n)$ complexity. Unnecessary instructions do not appear in the program, i.e., where $s_{i,j} = 0$ or duplicated register assignations. Slight modifying Algorithm 2 can produce instructions from decision variables with the same complexity. Figure 3 graphically shows the crossover, mutation, and repair process when performing over the decision variables.

### 2.5 Environmental Selection

The environmental selection is equivalent to the one found in traditional Genetic Algorithms (GAs). The current population and the offspring join to form a single population. Later, according to the fitness value of programs, only the $N$ best programs survive to the next generation, where $N$ is equal to the population size.

## 3 Experimental Configuration

This section gives the details to reproduce this chapter's experimentation. To evaluate our proposed Linear Genetic Programming (LGP) algorithm, we use the benchmark instances in [21] of 10 products with 25 stores (five small size instances), 25 products with 50 stores (five medium size instances), and 50 products with 100 stores (five large size instances).

Instances are available at https://github.com/AASantiago/ISHOP-U-Instances, and LGP source code is available at https://github.com/AASantiago/LGP-ISHOP-U/. This chapter LGP implementation uses the parameter settings from Table 1, a population size of 100, 25,000 as a maximum number of objective function evaluations, Binary Tournament as selection, 100% of crossover probability $p_r$ and a mutation probability $p_m$ of 5%. We reproduce the Genetic Algorithm (GA) described in [21], with their respective parameter settings for comparison purposes.

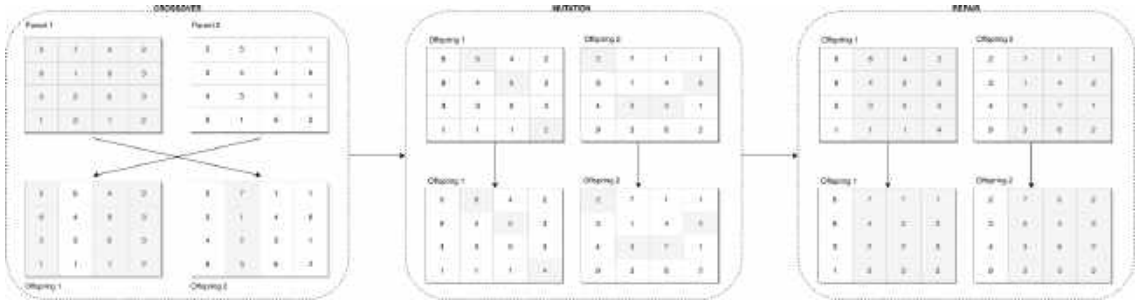


**Fig. 3.** Graphical representation of the evolutionary operators

**Table 2.** Experimental results for the GA and LGP over the considered 15 instances

| Problem | GA | LGP |
|---|---|---|
| UniformS1 | $447.2_0.0E0$ | $863.3_3.1E1$ |
| UniformS2 | $447.2_0.0E0$ | $1023.0_6.5E1$ |
| UniformS3 | $524.2_7.8E-1$ | $1103.6_8.5E1$ |
| UniformS4 | $462.9_0.0E0$ | $928.4_7.0E1$ |
| UniformS5 | $489.6_0.0E0$ | $955.1_8.2E1$ |
| UniformM1 | $3429.3_7.9E1$ | $6080.4_1.7E2$ |
| UniformM2 | $4721.1_1.2E2$ | $7644.1_3.1E2$ |
| UniformM3 | $5301.3_1.1E2$ | $8717.9_1.9E2$ |
| UniformM4 | $5068.5_1.1E2$ | $8299.3_2.5E2$ |
| UniformM5 | $5509.3_1.8E2$ | $9197.7_1.7E2$ |
| UniformL1 | $26161.2_5.8E2$ | $34419.5_1.2E3$ |
| UniformL2 | $21999.1_4.3E2$ | $29470.2_5.1E2$ |
| UniformL3 | $21361.1_3.8E2$ | $28811.2_5.4E2$ |
| UniformL4 | $23991.8_4.7E2$ | $32061.2_1.0E3$ |
| UniformL5 | $24036.8_4.0E2$ | $31671.4_8.1E2$ |

Due to the stochastic nature of the algorithms, we perform 30 independent runs over every considered ISHOP-U instance. To validate the statistical significance difference between the GA and LGP, we perform the Wilcoxon signed rank test [6], with an $\alpha = 0.05$ for a 95% of significance.

## 4 Results

This section outlines and discusses the numerical experimental results from the 30 independent executions of both algorithms in comparison GA and the LGP.

The experimentation numerical results are in Table 2 in terms of the achieved median value and interquartile range (IQR), with the format $MEDIAN_{IQR}$ and IQR in scientific notation. The median results are graphically show in Figure 2.

The instance name nomenclature starts with the distribution of the prices in the instances Uniform, followed by the instance size, S small, M Medium, L large, followed by their identification number; for example, UniformS1 is the instance number one of small size with a uniform distribution in their prices.

According to the results computed by the Wilcoxon signed rank test, a p-value$\leq 0.05$ is found in every considered instance, finding differences with a statistical significance of 95% between the GA and the LGP. The above difference favors the GA with a better achieved median value in all the instances, in the demerit of the LGP.

Given the computed numerical results, the Genetic Algorithm (GA) in [21] outperforms this chapter's Linear Genetic Programming proposal with statistical significance in the fifteen instances with uniform distribution prices in [21].

However, the purpose of this chapter is to prove that the concept of using Linear Genetic Programming (LGP) is feasible for the Internet Shopping Optimization Problem with multiple item Units purpose achieved. We highlight that LGP uses the same repair method as the GA, which is possible through an intermediate representation (the original problem decision variables) for later return to the instructions representation once feasible.

Our approach reutilizes evolutionary algorithm operators in evolutionary programming and can be applied to other problems straightforwardly. For future research, we want to design and implement new crossover and mutation operators, using the proposed mapping approach between decision variables and instructions for LGP to improve the results.

## Acknowledgments

Alejandro Santiago would like to thank CONAHCYT Mexico for the SNII salary award.

## References

1. **Aickelin, U., Dasgupta, D., Feng, G. (2013).** Artificial immune systems. Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, Springer US, Boston, MA, pp. 187–211. DOI: 10.1007/978-1-4614-6940-7\7.
2. **Amador-Larrea, S., Quiroz-Castellanos, M., Hoyos-Rivera, G., Mezura-Montes, E. (2022).** An experimental study of grouping crossover operators for the bin packing problem. Computación y Sistemas, Vol. 26, No. 2, pp. 663–682. DOI: 10.13053/cys-26-2-4249.
3. **Arellano-Verdejo, J., Guzmán-Arenas, A., Godoy-Calderon, S., Barrón-Fernández, R. (2014).** Efficiently finding the optimum number of clusters in a dataset with a new hybrid cellular evolutionary algorithm. Computación y Sistemas, Vol. 18, No. 2, pp. 313–327.
4. **Banzhaf, W., Nordin, P., Keller, R. E., Francone, F. D. (1998).** Genetic programming: an introduction: on the automatic evolution of computer programs and its applications. Morgan Kaufmann Publishers Inc. DOI: 10.5555/323917.
5. **Brameier, M., Banzhaf, W. (2007).** Linear genetic programming. Genetic and Evolutionary Computation, Springer US.
6. **Corder, G. W., Foreman, D. I. (2011).** Nonparametric statistics for non-statisticians: A step-by-step approach. Wiley. DOI: 10.1002/9781118165881.
7. **Dasgupta, D. (1999).** Artificial immune systems and their applications. Springer Berlin, Heidelberg.
8. **Dasgupta, D. (2006).** Advances in artificial immune systems. IEEE Computational Intelligence Magazine, Vol. 1, No. 4, pp. 40–49. DOI: 10.1109/MCI.2006.329705.
9. **Dasgupta, D., Yu, S., Nino, F. (2011).** Recent advances in artificial immune systems: Models and applications. Applied Soft Computing, Vol. 11, No. 2, pp. 1574–1587. DOI: 10.1016/j.asoc.2010.08.024.
10. **Eiben, A. E., Smith, J. E. (2015).** Introduction to evolutionary computing. Springer.
11. **Greensmith, J., Whitbrook, A., Aickelin, U. (2010).** Artificial immune systems. Handbook of Metaheuristics, pp. 421–448. DOI: 10.1007/978-1-4419-1665-5\14.
12. **Jenhani, I., Elouedi, Z. (2014).** Re-visiting the artificial immune recognition system: a survey and an improved version. Artificial Intelligence Review, Vol. 42, pp. 821–833. DOI: 10.1007/s10462-012-9360-0.
13. **Koza, J. R. (1992).** Genetic programming: On the programming of computers by means of natural selection. MIT Press.
14. **Koza, J. R. (1994).** Genetic programming II: Automatic Discovery of Reusable Programs. MIT Press.
15. **Koza, J. R. (1999).** Genetic programming III: Darwinian invention and problem solving. Morgan Kaufmann.
16. **Koza, J. R., Keane, M. A., Streeter, M. J., Mydlowec, W., Yu, J., Lanza, G. (2005).** Genetic programming IV: Routine human-competitive machine intelligence, Vol. 5, Springer New York, NY.
17. **Langdon, W. B., Poli, R., McPhee, N. F., Koza, J. R. (2008).** Genetic programming:

An introduction and tutorial, with a survey of techniques and applications. Computational Intelligence: A Compendium, pp. 927–1028. DOI: 10.1007/978-3-540-78293-3_22.

18. **Monroy, R., Saab, R., Godínez, F. (2004).** On modelling an immune system. Computación y Sistemas, Vol. 7, No. 4, pp. 249–259.
19. **Nebili, W., Farou, B., Kouahla, Z., Seridi, H. (2021).** Revised artificial immune recognition system. IEEE Access, Vol. 9, pp. 167477–167488. DOI: 10.1109/ACCESS.2021.3133731.
20. **Nimbiwal, M., Vashishtha, J. (2021).** A novel hybrid grey wolf optimization algorithm using two-phase crossover approach for feature selection and classification. Computación y Sistemas, Vol. 25, No. 4, pp. 793–801. DOI: 10.13053/cys-25-4-3931.
21. **Ornelas, F., Santiago, A., Martínez, S. I., Ponce-Flores, M. P., Terán-Villanueva, J. D., Balderas, F., Rocha, J. A. C., García, A. H., Laria-Menchaca, J., Treviño-Berrones, M. G. (2022).** The internet shopping optimization problem with multiple item units (ISHOP-U): Formulation, instances, np-completeness, and evolutionary optimization. Mathematics, Vol. 10, No. 14. DOI: 10.3390/math10142513.
22. **Poli, R., Langdon, W. B., McPhee, N. F. (2008).** A field guide to genetic programming.
23. **Rivera, G., Cisneros, L., Sánchez-Solís, P., Rangel-Valdez, N., Rodas-Osollo, J. (2020).** Genetic algorithm for scheduling optimization considering heterogeneous containers: A real-world case study. Axioms, Vol. 9, No. 1, pp. 27. DOI: 10.3390/axioms9010027.
24. **Rivera, G., Cruz-Reyes, L., Fernandez, E., Gomez-Santillan, C., Rangel-Valdez, N., Coello, C. A. C. (2023).** An ACO-based hyper-heuristic for sequencing many-objective evolutionary algorithms that consider different ways to incorporate the DM's preferences. Swarm and Evolutionary Computation, Vol. 76, pp. 101211. DOI: 10.1016/j.swevo.2022.101211.
25. **Romero-Ruiz, E., Segura, C. (2018).** Memetic algorithm with hungarian matching based crossover and diversity preservation. Computación y Sistemas, Vol. 22, No. 2, pp. 347–361. DOI: 10.13053/cys-22-2-2951.
26. **Singh, D. R., Singh, M. K., Singh, T., Prasad, R. (2018).** Genetic algorithm for solving multiple traveling salesmen problem using a new crossover and population generation. Computación y Sistemas, Vol. 22, No. 2, pp. 491–503. DOI: 10.13053/cys-22-2-2956.
27. **Timmis, J., Hone, A., Stibor, T., Clark, E. (2008).** Theoretical advances in artificial immune systems. Theoretical Computer Science, Vol. 403, No. 1, pp. 11–32. DOI: 10.1016/j.tcs.2008.02.011.
28. **Timmis, J., Knight, T., de-Castro, L. N., Hart, E. (2004).** An overview of artificial immune systems. Computation in Cells and Tissues: Perspectives and Tools of Thought, Springer Berlin Heidelberg, pp. 51–91. DOI: 10.1007/978-3-662-06369-9\4.
29. **Torres-Soto, A., Ponce-de-León-Sentí, E. E., Hernández-Aguirre, A., Torres-Soto, M. D., Díaz-Díaz, E. (2010).** Un sistema evolutivo robusto para la síntesis de circuitos analógicos. Computación y Sistemas, Vol. 13, No. 4, pp. 409–421.
30. **Watkins, A., Timmis, J., Boggess, L. (2004).** Artificial immune recognition system (AIRS): An immune-inspired supervised learning algorithm. Genetic Programming and Evolvable Machines, Vol. 5, pp. 291–317. DOI: 10.1023/B:GENP.0000030197.83685.94.
31. **Zheng, J., Chen, Y., Zhang, W. (2010).** A survey of artificial immune applications. Artificial Intelligence Review, Vol. 34, No. 1, pp. 19–34. DOI: 10.1007/s10462-010-9159-9.

*Article received on 28/02/2024; accepted on 15/05/2024.*
**Corresponding author is José Antonio Castán Rocha.*