Princess Sumaya University for Technology
جامعة الأميرة سميّة للتكنولوجيا

صندوق دعم البحث العلمي والابتكار
Scientific Research and Innovation Support Fund

Jordanian Journal of Computers and Information Technology

September 2023 VOLUME 09 NUMBER 03
ISSN 2415 - 1076 (Online)
ISSN 2413 - 9351 (Print)

JJCIT

www.jjcit.org jjcit@psut.edu.jo

An International Peer-Reviewed Scientific Journal Financed by the Scientific Research and Innovation Support Fund

Jordanian Journal of Computers and Information Technology (JJCIT)

The Jordanian Journal of Computers and Information Technology (JJCIT) is an international journal that publishes original, high-quality and cutting edge research papers on all aspects and technologies in ICT fields.

JJCIT is hosted and published by Princess Sumaya University for Technology (PSUT) and supported by the Scientific Research Support Fund in Jordan. Researchers have the right to read, print, distribute, search, download, copy or link to the full text of articles. JJCIT permits reproduction as long as the source is acknowledged.

AIMS AND SCOPE

The JJCIT aims to publish the most current developments in the form of original articles as well as review articles in all areas of Telecommunications, Computer Engineering and Information Technology and make them available to researchers worldwide. The JJCIT focuses on topics including, but not limited to: Computer Engineering & Communication Networks, Computer Science & Information Systems and Information Technology and Applications.

INDEXING

JJCIT is indexed in:

EDITORIAL BOARD SUPPORT TEAM

LANGUAGE EDITOR

Haydar Al-Momani

EDITORIAL BOARD SECRETARY

Eyad Al-Kouz

JJCIT ADDRESS

WEBSITE: www.jjcit.org
EMAIL: jjcit@psut.edu.jo
ADDRESS: Princess Sumaya University for Technology, Khalil Saket Street, Al-Jubaiha
B.O. BOX: 1438 Amman 11941 Jordan
TELEPHONE: +962-6-5359949
FAX: +962-6-7295534

LETTER TO THE EDITOR

Editor-In-Chief

Jordanian Journal of Computers and Information Technology

Dear Editor-In-Chief

Please accept my very best wishes and my heartfelt congratulations on the impressive and important work undertaken by you and your team at the Jordanian Journal of Computers and Information Technology. I am particularly proud of your recent achievement of Q2 in Scopus ranking which further highlights the journal's dedication to excellence in research and its impact on information technology in the Kingdom. It also underscores the importance of focusing on research in vital emerging areas such as 5G.

Indeed, I firmly believe that by supporting and publishing quality research on 5G technology, now and in the future, while also offering models and scenarios for the future of wireless communications in all aspects of life, including from 6G technology, your journal will support the pivotal work of researchers, industry, and academia. Open discussion and informed debate in your journal will help to address, preempt, and analyse the consequences of, and opportunities offered by, the continued demand for advanced features in wireless applications, including applications in smart cities that are underpinned by ultra-high speed internet capacity provided by 5G and future networks. Your journal can and should solidify its position as a leading platform for disseminating valuable insights in this rapidly evolving field. This is an undertaking that would certainly benefit the journal, but also the wider scientific community.

We can be in no doubt that the fifth generation of cellular technology is imminent and that its impact will be immense. However, this new technology is also in urgent need of a deep dive on relevance and rollout for government, industry, and society in our region. I have been impressed and reassured by the care and attention that is given by you to pressing technological and policy developments with widespread consequences for a range of stakeholders in our region – and not least those ordinary consumers and citizens whose lives are being altered immeasurably by an advancing tide of technological change. However, the 5G rollout is accelerating, and we must act fast to ensure that investment in cutting-edge research, policymaking and stewardship are conducted at a similar pace.

Despite the enormous potential benefits of 5G, we must also assess challenges and concerns from the outset. These include security and privacy issues, infrastructure

investment requirements, equality of access and representative data input, and, of course, environmental impacts. Indeed, we have learnt from those waves of technological change that we have all lived through in recent decades that it is essential to establish our own independent research investment sources and research publishing sources while harnessing the untold potential benefits of new technology in order to ensure successful and sustainable deployment.

By addressing concerns and leveraging our shared sense of mission, we may truly maximize the advantages of 5G, and help to unlock its full potential. 5G certainly has the potential to contribute trillions of dollars to global economic growth, to create new job opportunities, to drive innovation, to revolutionize healthcare and education, and to transform transportation systems. 5G technologies also offer a vast range of potential applications and offer the potential to address sustainability objectives while opening up new revenue streams for Telco's and others. The fifth generation of cellular technology will turbo-charge a digital transformation that has already changed our lives completely in little more than a decade. Our efforts to anticipate change and to manage impact will show us to be worthy stewards of a new world in which the lives of all our fellow citizens will be altered forever.

To conclude, I firmly believe that a commitment to, and focus on, cutting-edge research in 5G networks will prove essential to our research, policy, and implementation frameworks. I know that the Jordanian Journal of Computers and Information Technology must be central to this.

Thank you in anticipation.

Yours sincerely

HRH Princess Sumaya bint El Hassan

Chairman of the Board of Trustees

Princess Sumaya University for Technology

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

PREDICTION OF PEOPLE SENTIMENTS ON TWITTER USING MACHINE LEARNING CLASSIFIERS DURING RUSSIAN AGGRESSION IN UKRAINE

Mohammed Rashad Baker[1*], Yalmaz Najmaldeen Taher [2] and Kamal H. Jihad [3]

(Received: 12-Feb.-2023, Revised: 29-Apr.-2023, Accepted: 18-May-2023)

ABSTRACT

Social media has become an excellent way to discover people's thoughts about various topics and situations. In recent years, many studies have focused on social media during crises, including natural disasters or wars caused by individuals. This study examines how people expressed their feelings on Twitter during the Russian aggression on Ukraine. This study met two goals: the collected data was unique and it used Machine Learning (ML) to classify the tweets based on their effect on people's feelings. The first goal was to find the most relevant hashtags about aggression to locate the dataset. The second goal was to use several well-known ML models to organize the tweets into groups. The experimental results have shown that most of the performed ML classifiers have higher accuracy with a balanced dataset. However, the findings of the demonstrated experiments using data-balancing strategies would not necessarily indicate that all classes would perform better. Therefore, it is essential to highlight the importance of comparing and contrasting the data-balancing strategies employed in Sentiment Analysis (SA) and ML studies, including more classifiers and a more comprehensive range of use cases.

KEYWORDS

Sentiment analysis, Machine learning, Classification algorithm, Imbalanced data classification, Russian aggression in Ukraine.

1. INTRODUCTION

Crises have a major impact on human societies, altering the lives of individuals in significant ways. To understand the reactions of societies in times of crises, it is crucial to listen to people's ideas and comprehend their sentiments. Therefore, Sentiment Analysis (SA) has emerged as a vital subject of study in Natural Language Processing (NLP) and information extraction [1]. It seeks to evaluate a wide range of information, eliciting writers' emotions reflected in positive or negative words [2]. With the rise of social networking platforms, such as Facebook, Twitter, Linkedin and others, people have gained significant power in expressing and exchanging opinions about political or social events and inevitable social crises [3]. Nevertheless, understanding people's behaviors becomes challenging during crises because of the sheer volume of instructive messages, emotional outbursts, helpful safety suggestions and rumors. It is essential to leverage SA to better manage and regulate a crisis [4].

Natural disasters pose a significant challenge for societies and real-time sentiment analysis on social-media platforms (such as Twitter) can play a crucial role in saving lives [5]-[6]. Twitter's micro-blogging service allows users to share messages about events and news worldwide, using hashtags to follow hot topics. In the case of natural disasters, SA can be used to analyze tweets related to events, like the California Campfires (considered one of the most damaging and destructive wildfires in the history of California) [7]. However, there is a lack of research on the SA of natural disasters, causing negative impacts on society in many respects. More research attention and efforts need to study people's reactions to disasters. The studies must include mitigating, preparing, recovering and responding to disasters while reducing damage to citizens and economies [8]. During conflicts or aggressions caused by natural or human factors, opinions and sentiments can be expressed using social-media platforms, like Twitter [9].

Real-time assessment of public opinion expressed in tweets can aid authorities in developing early, response strategies. For example, a study examined the rule of the Taliban in Afghanistan after the

1. M. R. Baker is with Software Department, College of Computer Science and Information Technology, University of Kirkuk, Kirkuk, Iraq. Email: mohammed.rashad@uokirkuk.edu.iq
2. Y. N. Taher is with College of Computer Science and Information Technology, University of Kirkuk, Kirkuk, Iraq. Email: yalmaz.science@uokirkuk.edu.iq
3. K. H. Jihad is with College of Science, University of Kirkuk, Kirkuk, Iraq. Email: kamal.jihad@uokirkuk.edu.iq

withdrawal of US soldiers, using public opinion expressed in tweets [10]. Additionally, a new method was proposed for real-time sentiment analysis on the current Refugee Crisis to provide some prediction on polarity types of political improvement based on Twitter data [11]. Machine Learning (ML) techniques were used to present a method for sentiment analysis on Twitter data, comprising tweets about Afghanistan. The study identified algorithms and measures for evaluating the performance of supervised ML classifiers on tweets on the US war in Afghanistan [9]. The research focused on the refugee crisis. Accordingly, through binomial classification of positive and negative, ML algorithms were applied to obtain final-level decisions regarding the number of individuals commenting in support of refugees [11]. It is clear that exploring feelings has become critical, particularly in studying and processing natural languages [12]. More research is needed to leverage sentiment analysis to understand people's reactions during crises and develop strategies for effective crisis management.

The Russian aggression on Ukraine has raised questions about the formation and evolution of group identities during times of political tension [13]. Existing research suggests that insecurity, competition over resources and threat perception from out-groups increase ethnic-identity salience. However, Metzger et al. [14] proposed a novel approach using Ukrainian Twitter users' language preferences to examine this issue. The study found that key political events during the Ukrainian crisis did not lead to a reversion to language preferences, but following the annexation of Crimea, both Russian and Ukrainian Twitter users began using Russian tweets with greater frequency. Driscoll and Steinert-Threlkeld [15] suggested that social media provides insight into political attitudes and the study mapped the evolution of Russian-speaking communities' attitudes towards the conflict. The results show that the Russian-Ukrainian interstate border moved as far as the Russian military could advance without incurring occupation costs. These studies offer insights into the complex relationship between language, identity and political conflict and provide a basis for future research [14]-[15].

However, there has been comparatively less research into the social and emotional aspects of the Russian aggression in Ukraine, particularly the sentiments of those who are affected by it [16]. Understanding the sentiments of individuals towards aggression can provide several benefits. Firstly, it can provide insights into the effectiveness of various propaganda and messaging campaigns used by the parties involved, which can be useful in designing more effective messaging strategies. Secondly, it can provide insights into the emotional impact of aggression on individuals, which can help researchers better understand the psychological toll of aggression and identify areas where emotional support may be needed. Thirdly, sentiment analysis can identify potential sources of tension and aggression escalation. By monitoring sentiment trends over time, researchers can identify periods of heightened tension and negative sentiments that can signal the potential for further aggression or violence. Sentiment analysis of the Russian aggression in Ukraine is essential, as it can provide valuable insights into individuals' emotions, opinions and attitudes towards the aggression. It can be used to design more effective messaging strategies, provide emotional support to those affected by the aggression and identify potential sources of tension and aggression escalation.

ML techniques are important for sentiment analysis of the Russian aggression in Ukraine, because they enable the processing of large amounts of data quickly and accurately, allowing for a more comprehensive analysis of the sentiments expressed by individuals affected by the aggression. Additionally, ML algorithms can learn from the patterns and characteristics of the sentiment data to improve the accuracy of the sentiment analysis. This can provide valuable insights into individuals' emotions, opinions and attitudes toward aggression and can help identify potential sources of tension and aggression escalation. Moreover, ML techniques can be used to evaluate the effectiveness of different propaganda and messaging campaigns used by the parties involved, which can inform the design of more effective messaging strategies.

This study aims to predict people's sentiments on Twitter during Russian aggression [13] in Ukraine by using machine-learning classifiers while investigating how individuals respond and behave during a crisis, particularly in the context of a war or aggression. In this study, we aim to examine the potential of utilizing sentiment analysis using machine-learning techniques to comprehend community behavior and attitudes toward the war. It includes the degree to which the ML model can assist in comprehending community behavior, the level of correspondence between the observations and the actual user sentiments analyzed from the tweets, as well as the extent to which sentiments are uniform within and across regions. The major contributions of this study comprise the following:

1. A machine learning-based sentiment detection model for Twitter feeds concerning the war.
2. Using several machine-learning models for classifying sentiment polarity and emotions.
3. Intriguing insights into collective reactions to the war on social media could aid in informing decision-making processes and potentially contribute to developing more accurate and effective sentiment-analysis tools.

The structure of this article is as follows. Section 2 introduces the works related to this topic. In Section 3, we present our methodology. Section 4 discusses the experimental results. The conclusion will be drawn in Section 5.

2. RELATED WORKS

This section of the paper is organized into two sub-sections. The first sub-section provides a comprehensive literature review on sentiment analysis. The second sub-section focuses on related works on Russian aggression in Ukraine and how sentiment analysis has been used in this context.

2.1 Sentiment Analysis Related Works

SA has garnered significant interest in recent years due to the prominence of social-media sites, such as Twitter and Facebook. In addition, the availability of voluminous data in tweets, reviews and comments expedited its development. As a result, there is a substantial body of literature on SA [10]. The proposed method detects fake news using sentiments with positive and negative scores. Elmurngi and Gherbi [17] used statistical approaches to assess the efficacy of spambot systems in the SA arena. The task-specific precision of various ML models is tested. Wael et al. [18] used SA to identify Western media's bias in the Palestinian-Israeli crisis. This process includes finding deceptive terms, vocabularies and idioms used to sway public opinion about the Israel-Palestine problem.

The refugee issue was also considered utilizing SA. For instance, Ozturk and Ayvaz [19] analyzed Turkish and English tweets to address the challenges of Syrian refugees. They examined public feelings and opinions regarding the Syrian refugee situation. The results demonstrated a substantial variation in sentiments between Turkish and English tweets. The data also indicated that Turkish tweets include more optimistic sentiments. A comparative examination revealed that Turkish tweets contain more positive than negative or neutral sentiments toward Syrian refugees. Another considered issue was terrorism; for instance, Mansour [20] conducted SA on tweets related to ISIS to gain insights into how people feel about acts related to terrorism. The Term Frequency-Inverse Document Frequency (TF-IDF) approach was applied in the study to perform SA on tweets on ISIS.

Other researchers have used Twitter SA to determine various public opinions and feelings expressed during crises, such as civil wars and natural disasters [21]-[22]. Identifying these feelings is important for understanding the situations' dynamics and their emotional impact on affected people. A study shows that debriefing during a disaster can help authorities develop critical situational awareness and other programs to manage future events [23]. Studies showed that users' emotions fluctuate depending on location and proximity to the disaster site. For example, a study assessed the situation and public opinion regarding Brexit, in which more than 16 million tweets were collected. This study uncovered the most popular daily Twitter debates and discovered a positive correlation between Twitter's attitude towards Brexit and the British-pound exchange rate using the VADER library [24].

Protests have become more common in recent years and researchers are interested in understanding the emotions and sentiments expressed during these events through social media. Field et al. [25] used natural-language processing techniques to analyze emotions in tweets about the 2020 Black Lives Matter protests. They found that positive emotions, such as pride and hope, were prevalent in tweets with pro-BlackLivesMatter hashtags, contradicting stereotypical portrayals of protesters as perpetuating anger and outrage. Won et al. [26] developed a visual model that uses convolutional neural networks to classify the presence of protesters in an image and predict their visual attributes, perceived violence and exhibited emotions. They also released a novel dataset of 40,764 protest images with various annotations of visual attributes and sentiments. Steinert-Threlkeld and Joo [27] introduced the Multimodal Chile & Venezuela Protest Event Dataset (MMCHIVED), which contains city-day event data using a new source of data, text and images shared on social media, enabling the improved measurement of variables, such as protest size, protester and state violence, protesters' demographics and their emotions. Overall, these

studies demonstrate the value of analyzing social-media data to understand the emotions and sentiments expressed during protests.

According to the information reported above, SA has become a significant research topic in artificial intelligence. According to a survey of the available literature, various Twitter SA research employs classic ML algorithms to estimate sentiments from tweets [28]–[30]. These approaches tackle SA problems as if they were text-classification problems. These algorithms treat SA problems as text-classification problems and have been found to provide high accuracy with fewer computational resources. The classic ML algorithms commonly used for SA of Twitter data include Naive Bayes (NB), Random Forest (RF) and Logistic Regression (LR), among others. These algorithms provide strong accuracies with fewer computer resources and are used widely in SA of Twitter data [29].

In this work, we use a pre-annotated dataset using RoBERTa and TextBlob. Next, we apply various ML classifiers for SA to analyze tweets about the Russian-Ukrainian armed conflict. To the best of our knowledge, this research is among a few that seek to provide valuable insights into tweet content related to the Russian-Ukrainian war. Findings can be a reputable source of information to help governments and international organizations understand social-media trends and public views on the situation in Ukraine.

2.2 Russian Aggression in Ukraine Related works

Numerous studies have investigated the Russian aggression in Ukraine's online social networks (OSNs), focusing on Twitter and Reddit data to uncover hidden insights, disinformation campaigns and abnormal patterns [31]. There is a growing need for further research in the field, including aspect-based sentiment analysis (ABSA), to mine and analyze large datasets on OSNs. Hanley et al. [32] found differences in news coverage among Western, Russian and Chinese press outlets, with Russian media focusing on the purported justifications for the military operation and Chinese news media concentrating on the aggression's diplomatic and economic consequences. A novel lexicon-based unsupervised sentiment-analysis method was proposed by Guerra et al. [33] to measure "hope" and "fear" using Reddit.com as the main source of human reactions to daily events during nearly the first three months of the aggression.

Propaganda and misinformation were studied by Pierri et al. [34] on Facebook and Twitter during the first few months of the Russian aggression in Ukraine. They found that superspreaders played a disproportionate role in amplifying unreliable content and the political leaning of Facebook pages and Twitter users sharing propaganda was more right-leaning than average. In another study, Agarwal et al. [35] analyzed the emotional sentiments of tweets acquired during the peak war period, from December 31, 2021 to March 03, 2022. The study found more negative tweets than positive ones. It provided insights into the spread and influence of different categories of tweets, highlighting the need for further research on dynamic sentiment analysis.

In the study by Vyas et al. [36], a framework was developed to automatically classify distinct societal emotions related to the Russia-Ukraine War (RUW) on Twitter. The authors found that most tweets describe the RUW in key terms related more to Ukraine than to Russia and that 81% of Twitter users surveyed showed a neutral position toward the aggression. In another study, Vyas et al. [31] proposed a hybrid framework to automatically extract positive, negative and neutral sentiments from tweets related to the COVID-19 pandemic and classify them through machine-learning techniques.

Ibar-Alonso et al. [37] conducted a social-listening analysis on Twitter to assess sentiments and emotions regarding green energy during the onset of the 2022 Russian aggression in Ukraine. They found that the aggression changed society's sentiments about an energy transition to green energy, with negative feelings and emotions emerging in green-energy tweeters once the aggression started. The emotion of confidence increased as the aggression drove all countries to promote a rapid transition to greener-energy sources.

In the study by Chen et al. [38], the authors analyzed the public opinion warfare related to the Rural-Urban Waiver (RUW) in Chinese Weibo texts. They used Latent Dirichlet Allocation for unsupervised clustering and an opinion adversarial evolution algorithm to dynamically model the dominant degree of an opinion in the evolutionary processes. The authors released a dataset of Chinese Weibo associated with the RUW and proposed a data-driven approach for analyzing opinion warfare in cyber-physical-social systems. The study calls for further expansion of data collection and analysis from multiple

perspectives and the design of unsupervised clustering methods for complex social texts to improve opinion recognition.

Garcia and Cunanan-Yabut [39] analyzed the sentiments and emotions of the international community towards the Russian invasion of Ukraine using tweets posted on the first day in the #UkraineRussia hashtag. The results showed that negative sentiments were more prevalent and sadness was the most salient emotion. The study highlights the potential of social media, particularly Twitter, as a vehicle for mass communication that governments and politicians can use as a source of public opinion. Future research could continue examining the platform as a channel for public participation in peacemaking.

Benjamin Džubur et al. [40] combined sentiment and network analysis approaches to produce various insights into the discussion of the Russian aggression on Ukraine in their study. They discovered that most users support Ukraine and that the most critical accounts belong to political leaders, as well as relevant organizations or media outlets that actively report on the aggression. Apart from a few pro-Russia communities, all the groups express support for Ukraine to some degree. The study suggests that future research should focus on more thoughtful data collection and thorough analysis of various aspects of the networks.

3. METHODOLOGY

This section covers many methodological approaches that were taken throughout this research. Figure 1 shows that our methodology comprises four main steps. The first step is to examine the data-collection process and identify keywords. In the following step, pre-processing procedures have been applied to the dataset, starting with initial filtering and continuing to complete processing. After concluding the previous step, the next step addresses the topic of preparing the dataset for ML classification with a discussion that includes activities, such as the annotation technique and feature extraction. The last step applies the ML prediction models suggested for this research and discusses the results of the model performance.

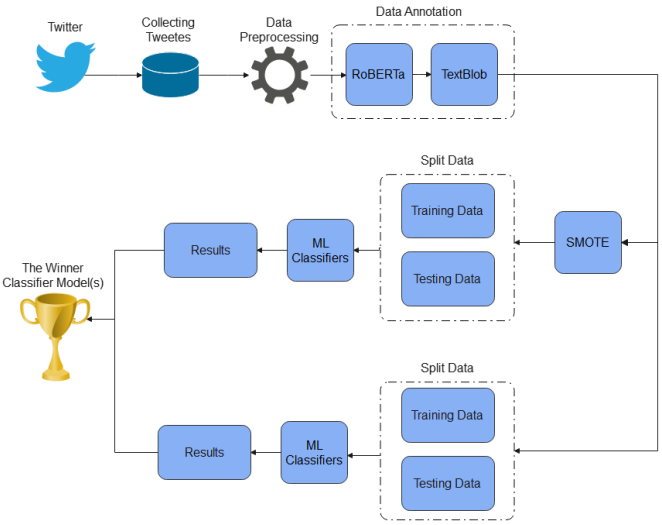

Figure 1. The steps of our methodology.

3.1 Data Collection

This research aims to comprehensively and representatively collect tweets about the aggression and crisis between Russia and Ukraine on Twitter. Prior research on aggression and crisis was conducted to identify the most relevant and popular data-collection hashtags. This involved an extensive review of news reports, social-media discussions and other relevant sources in identifying the key themes, topics and issues related to aggression.

Data collection for tweets related to aggression and crisis started on February 24, 2022 and lasted until July 4, 2022, on Twitter, which is the most popular platform for expressing thoughts and opinions [41]. The selected hashtags included "#UkraineVSRussia, #UkraineConflict, #UkraineCrisis, #UkraineWarCrimes, #stopwar, #UkraineWar, #UkraineRussia and #ukrainianwar," based on prior research and their popularity. The streaming Twitter API justifies its use, since it can give every tweet

for a given set of search terms [41]. However, Twint, a popular third-party tool, is also favored by researchers because of its flexibility and ability to scrape tweets without limitations or barriers [1]. We used the Twint project tool to scrape 1,245,791 tweets using these hashtags. We deliberately chose to use Twint over the Twitter API due to its ability to access all publicly available tweets, enabling historical-data retrieval that was crucial for our study. Twint also has advantages over the Twitter API, such as collecting data without authentication, avoiding rate limits and enabling advanced search queries. All tweets collected using Twint with these hashtags were saved as comma-separated values (.csv) for analysis in this research. To promote transparency and facilitate future research, we have made the dataset used in this manuscript publicly available to other researchers by uploading it to Mendeley [42].

3.2 Pre-processing

While tweets in other languages can provide insights into public sentiments during the ongoing crisis in Ukraine, our study focuses solely on English tweets. English is the most widely used language on Twitter, which ensured a substantial sample size for analysis. Furthermore, gathering and translating tweets in other languages would have required additional language processing, which was outside the scope of our study. Our dataset includes crisis-related keywords and was not restricted by location, resulting in tweets from all over the world. Before using Twitter's data for further processing, the data must undergo a filtering process to eliminate things with no relation. Several Python tools and libraries were used. The first step is to normalize the data to clean it up. As part of this process, non-English tweets, emojis, URLs and other special characters are filtered out. As a part of normalization, all capital letters are turned into small letters to simplify using text in the analysis [43]. The following step is tokenization. Tokenization is a pre-processing method that can divide any text into small pieces called "tokens." All special characters are taken out during tokenization, the spaces between words are found and abbreviations, numbers and other special characters are processed. The last step is removing English-language stop words like "is," "a," and "an," which do not add much to the text about a specific topic.

Figure 2 shows the distribution of the weekly tweets. The analysis indicates that the number of tweets per week varies significantly, ranging from a minimum of 677 to a maximum of 88778, with an average of 18112.3 tweets per week. This indicates that there is a lot of fluctuation in the amount of Twitter activity, which can be influenced by various factors, such as current events, popular topics and the behavior of users. Analyzing this data can help individuals and organizations better understand trends and patterns in Twitter activity and use this information to inform their social-media strategy.

Figure 3 displays a word cloud of the collected dataset related to the topic of Ukraine and Russia's conflict, with words, such as "destroy," "kill," "war," "force" and "military," alongside popular words, like "Ukraine," "Russia" and "War," suggesting a negative sentiment towards the conflict. The inclusion of words, like "peace," "help," "make," "child" and "live" in the word cloud could indicate a shift in sentiments towards a more positive direction. These words suggest the possibility of a desire for a resolution to the conflict and a focus on the impact of the conflict on civilians, particularly children and families.

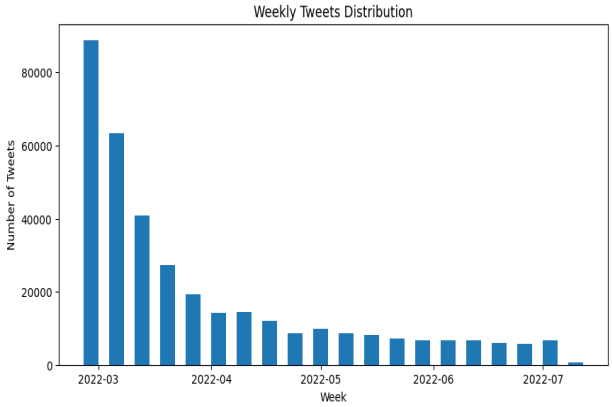

Figure 2. Weekly tweets distribution.

Figure 3. The word cloud generated from collected dataset.

Table 1 shows a list of 10 random tweets and their polarity scores generated after applying the RoBERTa model to perform sentiment analysis. The polarity scores indicate the sentiment expressed in the tweets, where a positive score denotes a positive sentiment and a negative score denotes a negative sentiment.

Table 1. List of 10 random tweets along with their polarity scores.

No.	Time	Tweet	Polarity
1	2022-02-25 15:50:54	foreign policy morality individuals nations act interest principles ordinary citizens force patriotism moral individuals spar nations moral scrutiny	-0.093
2	2022-02-26 22:13:04	miss Ukraine also join Ukrainian army let forget women fight beloved country	0.703
3	2022-02-27 07:26:20	watch Ukrainian news outlets make cry understand evil understand children die war	-1.02
4	2022-02-27 14:00:29	man find mine near Berdyansk pick hand cigarette mouth move away woods	0.101
5	2022-03-03 14:17:11	word speak furiously become cause unrest life war become cause destruction many generations	0.503
6	2022-03-08 02:05:02	deputy state Sherman say may become harder come days	-0.101
7	2022-03-10 15:05:21	experts keep say cannot faceoff Putin military Putin military commit war crimes already say cannot engage stink chamberlain criticism NATO us need make clear Russia	-0.033
8	2022-03-24 14:00:07	talk love ones Ukraine distress news via situation Ukraine crucial importance talk love ones supportive sensitive way	0.320
9	2022-04-30 02:44:43	Ukrainian girls consider one beautiful world today also defend country brave strong courageous different one thing unite desire win	0.576
10	2022-06-23 22:33:37	Ukraine receive long range rocket system Russian official threaten strike us embassy Kyiv	-0.025

A score of zero indicates a neutral sentiment. The table shows that the tweets cover various Ukraine-related topics, including foreign policy, military, news outlets, war crimes and personal relationships. The polarity scores of the tweets generated by RoBERTa and TextBlob vary slightly, indicating that different models may produce slightly different results depending on the text being analyzed. The tweet with the highest polarity score is number 2, expressing a positive sentiment towards Ukraine and its women who fight for their country. The tweet with the lowest polarity score is number 3, which expresses a highly negative sentiment towards Ukrainian news outlets and the reality of war. The other tweets have polarity scores that fall in between these two extremes, with some expressing positive sentiments (tweets 4, 5, 8 and 9), some expressing negative sentiments (tweets 1, 3, 6 and 10) and one tweet with neutral sentiment (tweet 7). Overall, the table provides a glimpse into the sentiments expressed on Twitter towards Ukraine during the time period covered by the tweets. The results demonstrate the usefulness of sentiment analysis in understanding public opinion and highlighting the importance of choosing the appropriate sentiment-analysis model for the specific context and purpose of the analysis.

3.3 Data Annotation

According to the research's main case for classification, the tweet annotation labels are assigned to various categories [44]. In our study, we fine-tuned the pre-trained Roberta model on a smaller annotated dataset relevant to our research domain. The annotated dataset consisted of text documents with labeled sentiment scores ranging from negative to positive. We used this dataset to train the model to identify and classify sentiments in the text data. Our study used the state-of-the-art RoBERTa model based on transformer architecture to perform sentiment analysis on text data for polarity. This allowed the model to learn contextual representations of words and sentences that could be fine-tuned for specific natural-language processing tasks. We fine-tuned the pre-trained Roberta model on a smaller annotated dataset consisting of text documents with labeled sentiment scores ranging from negative to positive, using the Hugging Face Transformers library to implement the PyTorch implementation of the model. To define the polarity, we used TextBlob on the result that we obtained from the RoBERTa transformer. During the model's training, we used a batch size of 32 and trained it for 10 epochs with a learning rate of 2e-5

while also using early stopping to prevent overfitting. Accordingly, the dataset was annotated into two categories for binary classification using TextBlob. The two categories are; negative tweets regarding the aggression annotated by 0 and positive tweets regarding the aggression annotated by 1. A total of 362,246 relevant tweets were considered after applying labeling to our collected dataset. The number of tweets in each of the two categories is depicted in Figure 4.

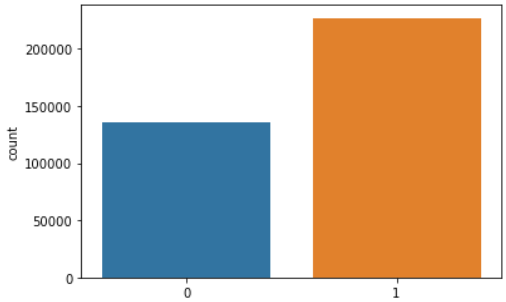

Figure 4. Distribution of categories in the dataset.

As we can notice from the figure, there are two classes of repented tweets. The first class, labeled 1, has 226,936 tweets about war and the second class, labeled 0, has 135,310 tweets about the same conversations during the aggression. The number of tweets in the first category is greater than in the second category, which could affect the results of the ML classifiers in the studied case. Synthetic Minority Oversampling Technique (SMOTE), a distribution-balancing technique, will be implemented to address this problem. Furthermore, classification findings will be shown before and after applying the SMOTE to establish how this technique improves categorization.

3.4 SMOTE

SMOTE is used in ML research for data balancing. Generating data samples of minority classification labels, such as the number of samples from each group, is nearly equal [45]. As described before, the dataset did not provide an equal distribution of categories, which can cause the ML models to overfit. To address the class imbalance in our dataset, we opted for the SMOTE, as it provides several benefits over other approaches. SMOTE is widely used in text classification and has proven effective in dealing with imbalanced datasets. One of its main advantages is that it generates synthetic examples for the minority class by interpolating between minority-class examples, reducing the risk of overfitting the training data.

Additionally, SMOTE produces synthetic examples similar to -but not identical to- the original minority-class examples, promoting diversity in the training data and improving the classifier's performance. SMOTE is also straightforward to implement and compatible with many classifiers. It is a desirable option for handling class imbalance in text-classification tasks that often have large and complex datasets.

3.5 Feature Extraction

In our work, we utilized both the unigram approach and the term frequency-inverse document frequency (TF-IDF) technique for feature extraction to highlight the sentiments of the tweets. The unigram approach is a simple, yet effective, method for capturing the essential words in a text corpus. Conversely, TF-IDF assigns weights to words based on their frequency in a document and their rarity in the corpus, which helps distinguish between common and rare words [43]. This technique is advantageous, as it can help identify the essential words in a text corpus and discard noise, resulting in a more meaningful and accurate representation of the data. Therefore, by combining both techniques, we could analyze the sentiment of the tweets in our dataset accurately.

For feature engineering, this research adopts TF-IDF. This approach operates by extracting weighted features from the data and assigning each data term with a few weight values into the model to enhance the performance of ML classifiers [46]. TF-IDF focuses on the most distinctive words, making its integration preferable to overcome the limitation of depending on word counts in SA research. Mathematical functions for TF-IDF are represented in Equations 1 and 2 as follows;

$$\mathrm{tf}(t,d) = \log\left(1 + f_{t,d}\right) \quad (1)$$

$$\mathrm{Idf}(t) = \log\left(\frac{1+N}{1+n_t}\right) \quad (2)$$

where tf(t,d) represents the count of term t in document d. N represents the total document number and n represents documents containing term t.

Our data is shuffled to make the classification performance more generalizable, reduce the variance and avoid model overfitting. The data is split into 80:20 ratio, where 80% is for training the model and 20% for testing it.

3.6 Classification Methods

In SA, ML classifiers have been utilized in diverse research groups for text classification. These various classifiers have provided different results depending on the applied case study. The ML model could be used for predicting what will happen in the future, learning something from the data or for both uses. First, a good training algorithm is needed to solve the optimization issues and store and process a considerable amount of data. Second, the representation and algorithm solutions for inference must be efficient and effective whenever the model has been properly trained and learned. A learning algorithm's reliability refers to how consistently it produces accurate results over time, even when presented with new data. A reliable algorithm can be trusted to make accurate predictions, even in situations where it has not encountered similar data before. In ML, it is not enough to have a model that can make accurate predictions; it is also important to have a model with a reliable learning algorithm that can continue to provide accurate results as new data becomes available. This is why it is essential to prioritize the reliability of the learning algorithm, even if it means sacrificing some computational resources, such as space and time [47].

All of the ML classifiers that have been covered up to this point produce incredible results across various scenarios, whether they involve SA or other ML context issues. Selecting the best classifier for a given scenario can be a complex and subjective process, as it depends on various factors, such as the size and nature of the dataset, the specific goals of the classification task and the desired number of classes. While a classifier with good performance may seem like the obvious choice, it is not always the case that its performance will remain consistent throughout the training process or when applied to different datasets.

Therefore, it may be necessary to consider multiple classifiers and evaluate their performance on the given dataset through experimentation and statistical analysis. A hypothesis could be formulated based on prior knowledge of the dataset or similar classification tasks, but ultimately, the best classifier must be determined empirically. It is important to note that there is no one-size-fits-all solution for selecting a classifier and the choice should be based on the specific requirements and constraints of the classification task. For that purpose, eight different classifiers will be evaluated side-by-side to check the most accurate one in solving the SA problem in this study. This sub-section will describe the primary classifiers utilized during this work. These classifiers include K-Nearest Neighbor (KNN), Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), Naïve Bayes (NB), XGBoost, AdaBoost and Multi-layer Perception (MLP) classifiers.

3.6.1 KNN

KNN is an essential ML classifier that uses instance-based learning. Text classification uses similarity measurements, which figure out how similar two points are by estimating their distance, proximity or clustering function [48]. In KNN, all training documents are saved and the calculations are postponed up to the classification stage [49]. KNN assigns a class based on the categories of the top neighbors of the labeled samples in the training set for each test document. The closer neighbors with the same category are, the more likely the prediction will be correct [50].

3.6.2 RF

RF is an ensemble classifier that uses bootstrapping and bagging to train several decision trees simultaneously [51]. When using an RF classifier, the final prediction is based on the most commonly observed class of objects and this method is called bagging [52]. A large number of predictors necessitates a lot of planted trees. Individual decision trees can be randomly decorated in various ways; for example, by selecting random features or data sub-sets [53]. To avoid overfitting problems that can

occur with individual decision trees because of their tremendous flexibility, RF uses many decision trees on a complex random sub-set of variables to create an effective solution [54].

3.6.3 DT

DT is an ML classification algorithm that works like a hierarchical tree. It utilizes attribute value constraints to split the training data into a few parts and uses different tests to display the tree branch. Each branch slope from the node matches the feature value. A DT works well, as the text-classification model does not have many features, but it is tough to make a classifier when there are a lot of features [55].

3.6.4 LR

LR is considered one of the most prevalent approaches to ML classification [56]. It does this by employing the concept of probability for a single test result by utilizing a logistic function in which the resulting probability might be either 1 or 0 [57]. This methodology has been implemented in various SA research studies [40]. For this reason, it is deemed to be one of the ML classifiers to be evaluated in the classification problem conducted by this research.

3.6.5 NB

NB algorithm is one of the most straightforward examples of a probabilistic classifier [58]. The training documents estimate a class-conditional document distribution, while Bayes' rule is used to get an estimate for test documents. The documents themselves are represented by their words. Furthermore, Naive Bayes may be better than discriminative classifiers for small sample sizes of data, because it has a built-in regularization that makes this method less likely to overfit [59].

3.6.6 XGBoost

Extreme gradient boosting, further called XGBoost, is a powerful ML algorithm that utilizes a gradient-boosting framework to train ensemble models. It works by iteratively adding decision trees to the ensemble, with each new tree correcting the errors of the previous ones, ultimately leading to a more accurate prediction [60]. XGBoost also includes several regularization techniques to prevent overfitting and improve model performance.

3.6.7 AdaBoost

AdaBoost is the first functional boosting classifier suggested by Freund [61]. It combines multiple base classifiers, usually decision trees, to build an accurate classifier. It invokes a weak classifier and provides various training data distributions for each call. The classifier can remove unnecessary features in the training data so that important features are used in the training process. AdaBoost has been utilized in various application studies [62] and has been deemed suitable for the comparative results of this study.

3.6.8 MLP

The MLP is an artificial neural-network architecture, which is probably the most widely used for classification and regression today [63]. MLPs are feed-forward neural networks usually made up of several layers of nodes that only connect in one direction and are usually trained by backpropagation [64].

3.7 Performance Metrics

The performance evaluation measures that are discussed in this research include, Accuracy (Acc.), Precision (Pr.), Recall (Re.), F1 score and Matthews Correlation Coefficient (MCC). These metrics are defined as follows:

$$Accuracy\ (Acc.) = \frac{TP+TN}{TP+FN+FP+TN} \tag{3}$$

$$Precision\ (Pr.) = \frac{TP}{TP+FP} \tag{4}$$

$$Recall\ (Re.) = \frac{TP}{TP+FN} \tag{5}$$

$$F1\ Score = 2x\ \frac{Pr.x\ Re.}{Pr.+Re.} \tag{6}$$

$$MCC = \frac{TP\ x\ TN - FP\ x\ FN}{\sqrt{(TP+FP)x\ (TP+FN)x(TN+FP)x(TN+FN)}} \tag{7}$$

TP, FP, TN and FN stand for true positive, false positive, true negative and false negative, respectively.

4. EXPERIMENTAL RESULTS

All classifiers were subjected to two separate experiments: in the first one, the data was imbalanced and in the second one, the imbalanced data was addressed and handled using SMOTE. Table 2 shows the evaluation metrics, including accuracy, precision, recall, F-score and MCC, before applying SMOTE.

Table 2. Results of ML models before applying SMOTE.

Model	Class	Precision	Recall	F-Score	Accuracy	MCC
KNN	0	0.76	0.69	0.72	0.80	0.571
	1	0.83	0.87	0.85		
	Macro avg.	0.79	0.78	0.78		
RF	0	0.95	0.92	0.93	0.95	0.894
	1	0.95	0.97	0.96		
	Macro avg.	0.95	0.94	0.95		
DT	0	0.89	0.92	0.90	0.93	0.847
	1	0.95	0.94	0.94		
	Macro avg.	0.92	0.93	0.92		
LR	0	0.96	0.95	0.95	0.97	0.928
	1	0.97	0.98	0.97		
	Macro avg.	0.97	0.96	0.96		
NB	0	0.85	0.86	0.85	0.89	0.762
	1	0.91	0.91	0.91		
	Macro avg.	0.88	0.88	0.88		
XGBoost	0	0.93	0.83	0.88	0.91	0.816
	1	0.91	0.96	0.93		
	Macro avg.	0.92	0.90	0.91		
AdaBoost	0	0.88	0.54	0.67	0.80	0.572
	1	0.78	0.96	0.96		
	Macro avg.	0.83	0.75	0.76		
MLP	0	0.97	0.97	0.97	0.98	0.956
	1	0.98	0.98	0.98		
	Macro avg.	0.98	0.98	0.98		

In the first experiment, it can be shown that MLP and LR performed superiorly to all of the other ML classifiers in terms of accuracy, with scores of 0.98 and 0.97, respectively. Besides that, RF, DT and XGBoost followed with scores of 0.95, 0.93 and 0.91, correspondingly. Regarding the models on the left side, NB obtained the highest accuracy, which was 0.89, followed by KNN and AdaBoost, with the lowest accuracy of 0.80. In addition to accuracy, the MCC has been recognized in the literature as a comprehensive performance evaluation for binary-classification issues, especially true when using imbalanced and balanced datasets as an evaluation criterion. In this regard, the MCC scored the most for MLP with a value of 0.956, followed by LR with 0.928. However, the score had the lowest for AdaBoost and KNN, with values of 0.572 and 0.571, respectively.

Next, the same classifiers were applied again after balancing the distributed dataset using SMOTE. This experiment was carried out to demonstrate how SMOTE can improve the performance of classifiers after they have been applied to an imbalanced dataset. As they are involved here, earlier employed evaluation measures can also be found in Table 3.

Table 3. Results of ML models after applying SMOTE.

Model	Class	Precision	Recall	F-Score	Accuracy	MCC
KNN	0	0.62	0.99	0.76	0.69	0.423
	1	0.97	0.40	0.57		
	Macro avg.	0.79	0.69	0.66		
RF	0	0.95	0.96	0.96	0.96	0.910
	1	0.96	0.95	0.96		
	Macro avg.	0.96	0.96	0.96		
DT	0	0.95	0.94	0.94	0.94	0.884
	1	0.94	0.95	0.94		
	Macro avg.	0.94	0.94	0.94		
LR	0	0.97	0.97	0.97	0.97	0.944
	1	0.97	0.97	0.97		
	Macro avg.	0.97	0.97	0.97		
NB	0	0.86	0.91	0.89	0.88	0.767
	1	0.91	0.85	0.88		
	Macro avg.	0.88	0.88	0.88		
XGBoost	0	0.95	0.90	0.92	0.93	0.853
	1	0.90	0.95	0.93		
	Macro avg.	0.93	0.93	0.93		
AdaBoost	0	0.92	0.59	0.72	0.77	0.577
	1	0.70	0.95	0.80		
	Macro avg.	0.81	0.77	0.76		
MLP	0	0.99	0.98	0.99	0.99	0.970
	1	0.98	0.99	0.98		
	Macro avg.	0.99	0.99	0.99		

For the following experiment, it can be observed that the best performance was attributed to MLP with 0.99 accuracy, followed by LR, RF, then DT with accuracies of 0.97, 0.96 and 0.94, respectively. Worst accuracy performance was attributed to AdaBoost with 0.77, then KNN with 0.69. As for MCC results, MLP was the highest classifier with 0.97, followed by LR with 0.944, then RF with 0.910. The worst MCC performance was observed at 0.423 in KNN classifier. It is indicated from the results that some classifiers' accuracies have improved after SMOTE was applied to the imbalanced dataset. At the same time, some classifiers' performance has degraded. Still, it is confirmed that MCC across all classifiers has improved, which shows the suitability of SMOTE in performance evaluation after balancing the dataset.

4.1 Comparative Analysis

This sub-section compares the accuracy and MCC values of the results for all ML classifiers before and after using SMOTE. The comparison is illustrated in Figures 5 and 6, respectively.

It is observed that accuracy and MCC are among the most important measures used to evaluate the performance of ML classifiers. Based on the analysis and the analyzed case in this research, it is evident that when the SMOTE technique was applied, the performance of four of the classifiers increased: RF from 0.95 to 0.96, DT from 0.93 to 0.94, XGBoost from 0.91 to 0.93 and MLP from 0.98 to 0.99. With 0.97, only LR kept its accuracy before and after SMOTE. However, the remaining three ML classifiers, Adaboost, KNN and NB, did not demonstrate any gain in accuracy. These results indicate the suitability of the SMOTE technique in terms of accuracy. However, another important measure, MCC introduced in the literature, is more robust and trustworthy than balanced accuracy in F1 score and binary classification analysis [65]. The MCC data shows that most classifiers exhibited an increase after implementing SMOTE, with the most significant improvement reported for MLP 0.97 MCC score, followed by LR 0.944 MCC score. The MCC scores for RF, DT and XGBoost are 0.91, 0.884 and 0.853, respectively. Only the AdaBoost classifier showed a minor gain in the MCC score, bringing it to 0.577. However, the MCC score of the KNN classifier decreased after applying SMOTE, which is also consistent with accuracy.

Figure 7 shows the ROC curve analysis for applied ML models. The results show that MLP, LR and RF have the highest AUC-ROC values, with values of 0.987, 0.988 and 0.987, respectively. These results

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

indicate that these models are more accurate and reliable in predicting the target variable. The KNN model has an AUC-ROC value of 0.863, lower than the other three top-performing models. This suggests that the KNN model may not perform as well in specific scenarios where the other models are better suited. The NB and DT models have lower AUC-ROC values of 0.936 and 0.927, respectively. These values are lower than for the top-performing models, indicating that these models may not be as accurate in predicting the target variable as the others. Lastly, Adaboost and XGB models have AUC-ROC values of 0.865 and 0.965, respectively. While the AUC-ROC value for Adaboost is lower than most other models, the XGB model has a relatively high AUC-ROC value, suggesting that it may be a good alternative to the top-performing models.

Overall, the AUC-ROC values give a good indication of the relative performance of each classification model and can be used to guide our choice of the best model for our specific classification problem.

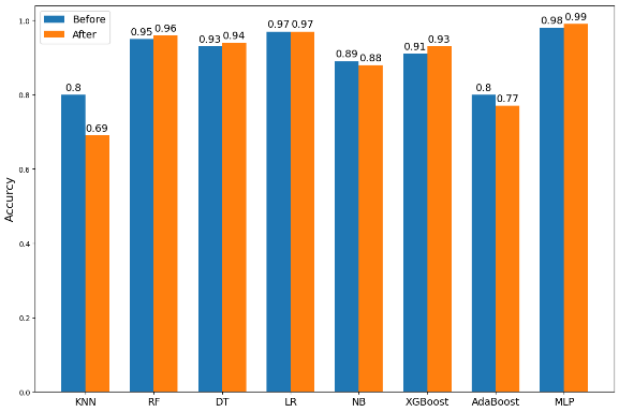

Figure 5. Comparative analysis of accuracy before and after applying SMOTE.

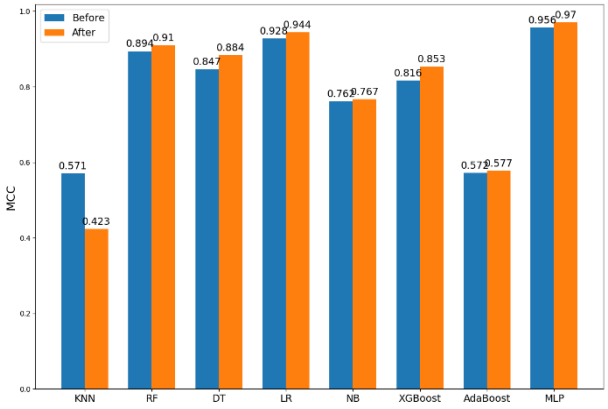

Figure 6. Comparative analysis of MCC before and after applying SMOTE.

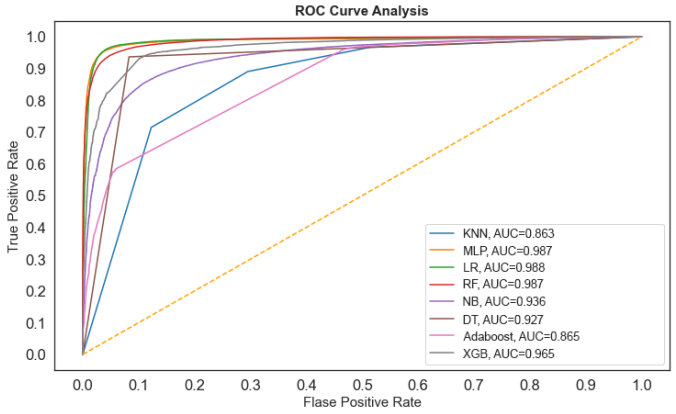

Figure 7. ROC curve analysis for applied ML models.

4.2 Result Discussion

In the classification of Russian aggression in Ukraine-related discussion on Twitter, it is evident that most basic ML classifiers improved their performance, which was confirmed by measuring the MCC score as identified in the literature to be one of the best approaches for classification problems, particularly when data is balanced utilizing techniques, such as SMOTE. The only classifier not enhanced by the used approach was the KNN classifier, validated by the MCC score and the accuracy result. Even so, the KNN algorithm performed far higher when the data was imbalanced than when the data was balanced. This demonstrates that despite the promise of data-balancing methodologies, their application in producing a balanced dataset could not always be applicable across all ML classifiers. As a result, it is worthwhile to investigate the possibility of determining the performance of these various classifiers by employing additional data-balancing methods to evaluate and compare their performance.

In this study, the numerical results obtained from the ML models should be discussed to provide insights into the performance of the models. The results presented in Table 1 and Table 2 show the performance of the models before and after applying SMOTE, respectively. Before applying SMOTE, the RF and

MLP models had the highest precision, recall, F-score, accuracy and MCC values, indicating that they performed the best among the models. After applying SMOTE, the RF and MLP models still had the highest values for most of these metrics, indicating that they continued to perform well even after the dataset was balanced using SMOTE.

In contrast, the KNN and AdaBoost models had lower performances before applying SMOTE, with lower precision, recall, F-score, accuracy and MCC values. After using SMOTE, these models showed some performance improvement, but did not perform as well as the RF and MLP models. The DT and NB models had moderate performances before and after applying SMOTE, with relatively consistent values for most metrics. These results can be discussed regarding the strengths and weaknesses of the different models and how well they handled the imbalanced dataset. Additionally, the implications of these results for the problem being addressed in the study can be discussed, including any recommendations for selecting a model or improving the performances of the models.

Table 4. Comparison with existing Twitter sentiment classification methods.

Reference	**Twitter Sentiment Classifier**	**Accuracy (%)**
[66] 2018	RNN-Capsule	91.6%
[67] 2019	Hybrid CNN-LSTM model	91%
[68] 2021	ConvBiLSTM model	91.3%
Our best model	MLP	98%

Table 4 provides a comparison of the performance of the proposed MLP model with those of three existing Twitter sentiment classification methods. The table reports the accuracy of each method as a percentage. It can be observed that the proposed MLP model outperforms the existing methods, achieving an accuracy of 98%. In contrast, the existing methods report 91% to 91.6% accuracy. The comparison of the proposed MLP model with the current methods demonstrates the effectiveness of the proposed approach in classifying Twitter sentiments. The MLP model outperforms the existing methods by a significant margin, indicating that the proposed approach can improve the accuracy of Twitter sentiment classification. However, it is essential to note that the comparison is limited to the reported accuracy metric. Other evaluation metrics, such as precision, recall, F1 score and ROC-AUC, should also be considered to evaluate the proposed method comprehensively. Overall, the results in Table 4 suggest that the proposed MLP model is a promising approach for Twitter sentiment classification and can provide improved accuracy compared to existing methods.

5. CONCLUSION

In this study, sentiments about war during the Russian aggression in Ukraine have been analyzed. This study achieved two goals: the uniqueness of the collected data and ML to categorize the tweets' sentiments. The first goal was to collect the dataset by searching for the most popular hashtags about aggression. The second goal was to place the collected tweets into categories using several well-known ML models. The most basic ML classifiers improved their performance, confirmed by evaluating the MCC score, which is known in the literature as one of the best ways to solve classification problems, especially when data is balanced using techniques like SMOTE. Also, it was demonstrated that data-balancing techniques would not guarantee that all classes could perform better. Nevertheless, the data-balancing approach must be tested and compared using different ML classifiers and SA evaluation datasets.

The prediction of sentiment analysis on Russian aggression in Ukraine using ML models has significant implications for the academia. Firstly, it can enhance our understanding of the social and emotional aspects of aggression, particularly the sentiments of those affected by it. By predicting sentiment trends over time, researchers can identify patterns in public opinion and gain insights into the underlying causes and factors that contribute to positive or negative sentiments.

Secondly, using ML models for sentiment analysis can provide a more accurate and efficient analysis of large volumes of data. Traditional manual sentiment-analysis methods can be time-consuming and subjective, leading to potential biases and errors. ML models, on the other hand, can analyze large datasets in real time, providing quick and accurate results.

Furthermore, using ML models for sentiment analysis can provide valuable insights for policymakers

and decision-makers. Policymakers can develop more effective conflict-resolution and peace-building strategies by identifying potential sources of tension and aggression escalation. Moreover, ML models can help determine the effectiveness of propaganda and messaging campaigns used by the parties involved, which can aid in designing more effective messaging strategies.

On the other hand, there are several limitations to consider in the study. Firstly, shortened links and multimedia content were not considered, leading to underestimating Russian propaganda and other sources. Secondly, the study relied on a distant-supervision approach rather than manual verification, which could introduce errors and biases. Additionally, the method used to assess the amount of removed content was imperfect and did not allow for the exact reasons for removal. The study did not account for the activity of automated accounts that could spread misinformation.

In summary, the prediction of sentiment analysis on Russian aggression in Ukraine using ML models can advance our understanding of the conflict's social and emotional aspects, provide an accurate and efficient analysis of large volumes of data and aid policymakers in developing more effective conflict-resolution strategies. As such, it is an important area of research for the academia.

In the future, the current research can be expanded by incorporating deep-learning classifiers, exploring various feature settings, experimenting with different data-balancing techniques and conducting more predictive analysis research on both the SA dataset presented here and other benchmarking datasets from the research literature. These future works have the potential to enhance our technical understanding of ML and its configurations and parameters and provide us with deeper insights into the performance of ML models in sentiment-analysis tasks. By further exploring these avenues, we can better understand the strengths and limitations of different ML algorithms and techniques and identify more effective ways to optimize their performance. Overall, these future works will help advance the field of ML and its application in sentiment analysis and open up new avenues for future research.

COMPLIANCE WITH ETHICAL STANDARDS

We certify that our work is original and does not plagiarize the work of others.

COMPETING INTERESTS

We certify that there is no actual or potential conflict of interest in relation to this article.

RESEARCH DATA POLICY AND DATA AVAILABILITY STATEMENTS

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

ACKNOWLEDGEMENTS

We would like to thank Dr. Ahmed M. Fakhrudeen from the University of Kirkuk for his invaluable contribution to this manuscript. His efforts in proofreading and providing feedback have significantly improved its quality. We appreciate his dedication to the field and extend our heartfelt thanks for his valuable support in bringing this manuscript to fruition.

REFERENCES

[1] A. H. Alamoodi, M. R. Baker, O. S. Albahri, B. B. Zaidan and A. A. Zaidan, "Public Sentiment Analysis and Topic Modeling Regarding COVID-19's Three Waves of Total Lockdown: A Case Study on Movement Control Order in Malaysia," KSII Trans. Internet Inf. Syst., vol. 16, no. 7, pp. 2169–2190, DOI: 10.3837/tiis.2022.07.003, 2022.

[2] N. Afroz, M. Boral, V. Sharma and M. Gupta, "Sentiment Analysis of COVID-19 Nationwide Lockdown Effect in India," Proc. of the Int. Conf. on Artificial Intelligence and Smart Systems (ICAIS 2021), pp. 561–567, DOI: 10.1109/ICAIS50930.2021.9396038, 2021.

[3] S. Hajrahnur, M. Nasrun, C. Setianingsih and M. A. Murti, "Classification of Posts on Twitter Traffic Jam in the City of Jakarta Using Algorithm C4.5," Proc. of the 2018 Int. Conf. on Signals and Systems (ICSigSys 2018), pp. 294–300, DOI: 10.1109/ICSIGSYS.2018.8372776, 2018.

[4] P. Kostakos, M. Nykanen, M. Martinviita, A. Pandya and M. Oussalah, "Meta-terrorism: Identifying Linguistic Patterns in Public Discourse After an Attack," Proc. of the 2018 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining (ASONAM 2018), pp. 1079–1083, DOI:

10.1109/ASONAM.2018.8508647, 2018.
[5] G. M. Demirci, S. R. Keskin and G. Dogan, "Sentiment Analysis in Turkish with Deep Learning," Proc. of the 2019 IEEE Int. Conf. on Big Data (Big Data 2019), pp. 2215–2221, DOI: 10.1109/BigData47090.2019.9006066, 2019.
[6] J. P. Singh, Y. K. Dwivedi, N. P. Rana, A. Kumar and K. K. Kapoor, "Event Classification and Location Prediction from Tweets during Disasters," Annals of Operations Research, vol. 283, no. 1–2, pp. 737–757, DOI: 10.1007/s10479-017-2522-3, Dec. 2019.
[7] N. H. Khun, T. T. Zin, M. Yokota and H. A. Thant, "Emotion Analysis of Twitter Users on Natural Disasters," Proc. of the 2019 IEEE 8th Global Conf, on Consumer Electronics (GCCE 2019), pp. 342–343, DOI: 10.1109/GCCE46687.2019.9015234, 2019.
[8] U. H. H. Zaki, R. Ibrahim, S. A. Halim, K. A. M. Khaidzir and T. Yokoi, "Sentiflood: Process Model for Flood Disaster Sentiment Analysis," Proc. of the 2017 IEEE Conf. on Big Data and Analytics (ICBDA 2017), vol. 2018-Janua., pp. 37–42, DOI: 10.1109/ICBDAA.2017.8284104, 2018.
[9] S. K. Akpatsa et al., "Sentiment Analysis and Topic Modeling of Twitter Data: A Text Mining Approach to the US-Afghan War Crisis," SSRN Electronic J., DOI: 10.2139/ssrn.4064560, 2022.
[10] E. Lee, F. Rustam, I. Ashraf, P. B. Washington, M. Narra and R. Shafique, "Inquest of Current Situation in Afghanistan under Taliban Rule Using Sentiment Analysis and Volume Analysis," IEEE Access, vol. 10, pp. 10333–10348, DOI: 10.1109/ACCESS.2022.3144659, 2022.
[11] M. Mahiuddin, "Real Time Sentiment Analysis and Opinion Mining on Refugee Crisis," Proc. of the 2019 5th Int. Conf. on Advances in Electrical Engineering (ICAEE 2019), pp. 699–705, DOI: 10.1109/ICAEE48663.2019.8975462, 2019.
[12] A. Alamoodi et al., "Sentiment Analysis and Its Applications in Fighting COVID-19 and Infectious Diseases: A Systematic Review," Expert Systems with Applications, vol. 167, p. 114155, 2020.
[13] G. Assembly, "Aggression against Ukraine: Resolution / Adopted by the General Assembly," United Nations, [Online], Available: https://digitallibrary.un.org/record/3959039?ln=en, 2022.
[14] M. M. Metzger, R. Bonneau, J. Nagler and J. A. Tucker, "Tweeting Identity? Ukrainian, Russian and# Euromaidan," J. of Comparative Economics, vol. 4, no. 1, pp. 16–40, 2016.
[15] J. Driscoll and Z. C. Steinert-Threlkeld, "Social Media and Russian Territorial Irredentism: Some Facts and a Conjecture," Post-Soviet Aff., vol. 36, no. 2, pp. 101–121, Mar. 2020.
[16] R. A. Bryant, P. P. Schnurr and D. Pedlar, "Addressing the Mental Health Needs of Civilian Combatants in Ukraine," The Lancet Psychiatry, vol. 9, no. 5, pp. 346–347, 2022.
[17] E. Elmurngi and A. Gherbi, "Detecting Fake Reviews through Sentiment Analysis Using Machine Learning Techniques," Proc. of the 6th Int. Conf. Data Analytics Detection (DATA Anal. 2017), no. c, pp. 65–72, 2017.
[18] W. F. Al-Sarraj and H. M. Lubbad, "Bias Detection of Palestinian/Israeli Conflict in Western Media: A Sentiment Analysis Experimental Study," Proc. of the 2018 Int. Conf. on Promising Electronic Technologies (ICPET 2018), pp. 98–103, DOI: 10.1109/ICPET.2018.00024, 2018.
[19] N. Öztürk and S. Ayvaz, "Sentiment Analysis on Twitter: A Text Mining Approach to the Syrian Refugee Crisis," Telematics and Informatics, vol. 35, no. 1, pp. 136–147, DOI: 10.1016/j.tele.2017.10.006, 2018.
[20] S. Mansour, "Social Media Analysis of Users' Responses to Terrorism Using Sentiment Analysis and Text Mining," Procedia Computer Science, vol. 140, pp. 95–103, DOI: 10.1016/j.procs.2018.10.297, 2018.
[21] G. A. Ruz, P. A. Henríquez and A. Mascareño, "Sentiment Analysis of Twitter Data During Critical Events through Bayesian Networks Classifiers," Future Generation Computer Systems, vol. 106, pp. 92–104, DOI: 10.1016/j.future.2020.01.005, 2020.
[22] F. Yao and Y. Wang, "Domain-specific Sentiment Analysis for Tweets during Hurricanes (DSSA-H): A Domain-adversarial Neural-network-based Approach," Computers, Environment and Urban Systems, vol. 83, DOI: 10.1016/j.compenvurbsys.2020.101522, 2020.
[23] A. Squicciarini, A. Tapia and S. Stehle, "Sentiment Analysis during Hurricane Sandy in Emergency Response," Int. J. of Disaster Risk Reduct., vol. 21, pp. 213–222, DOI: 10.1016/j.ijdrr.2016.12.011, 2017.
[24] S. H. W. Ilyas, Z. T. Soomro, A. Anwar, H. Shahzad and U. Yaqub, "Analyzing Brexit's Impact Using Sentiment Analysis and Topic Modeling on Twitter Discussion, " Proc. of the ACM Int. Conf., pp. 1–6, DOI: 10.1145/3396956.3396973, Jun. 2020.
[25] A. Field, C. Y. Park, A. Theophilo, J. Watson-Daniels and Y. Tsvetkov, "An Analysis of Emotions and the Prominence of Positivity in #BlackLivesMatter Tweets," Proc. of the National Academy of Sciences of the United States of America, vol. 119, no. 35, p. e2205767119, DOI: 10.1073/pnas.2205767119, 2022.
[26] D. Won, Z. C. Steinert-Threlkeld and J. Joo, "Protest Activity Detection and Perceived Violence Estimation from Social Media Images," Proc. of the 2017 ACM Multimedia Conf.e (MM 2017), pp. 786–794, DOI: 10.1145/3123266.3123282, Oct. 2017.
[27] Z. Steinert-Threlkeld and J. Joo, "MMCHIVED: Multimodal Chile and Venezuela Protest Event Data," Proc. of the 16th Int. AAAI Conf. on Web and Social Media, vol. 16, pp. 1332–1341, DOI: 10.1609/icwsm.v16i1.19385, 2022.
[28] F. Rustam, M. Khalid, W. Aslam, V. Rupapara, A. Mehmood and G. S. Choi, "A Performance Comparison

of Supervised Machine Learning Models for Covid-19 Tweets Sentiment Analysis," PLoS One, vol. 16, no. 2, DOI: 10.1371/journal.pone.0245909, Feb. 2021.

[29] Imamah and F. H. Rachman, "Twitter Sentiment Analysis of Covid-19 Using Term Weighting TF-IDF and Logistic Regresion," Proc. of the 6th Information Technology Int. Seminar (ITIS 2020), pp. 238–242, DOI: 10.1109/ITIS50118.2020.9320958, 2020.

[30] P. Sharma and A. K. Sharma, "Experimental Investigation of Automated System for Twitter Sentiment Analysis to Predict the Public Emotions Using Machine Learning Algorithms," Materials Today Proc., DOI: 10.1016/j.matpr.2020.09.351, 2020.

[31] M. Caprolu, A. Sadighian and R. Di Pietro, "Characterizing the 2022 Russo-Ukrainian Conflict through the Lenses of Aspect-based Sentiment Analysis: Dataset, Methodology and Preliminary Findings," arXiv Prepr, arXiv2208.04903, [Online], Available: http://arxiv.org/abs/2208.04903, Aug. 2022.

[32] H. W. A. Hanley, D. Kumar and Z. Durumeric, "A Special Operation': A Quantitative Approach to Dissecting and Comparing Different Media Ecosystems' Coverage of the Russo-Ukrainian War," arXiv Prepr, arXiv 2210.03016, [Online], Available: https://doi.org/10.48550/arXiv.2210.03016, Oct. 2022.

[33] A. Guerra and O. Karakuş, "Sentiment Analysis for Measuring Hope and Fear from Reddit Posts during the 2022 Russo-Ukrainian Conflict," arXiv Prepr, arXiv2301.08347, [Online], Available: http://arxiv.org/abs/2301.08347, Jan. 2023.

[34] F. Pierri, L. Luceri, N. Jindal and E. Ferrara, "Propaganda and Misinformation on Facebook and Twitter during the Russian Invasion of Ukraine," arXiv Prepr, arXiv2212.00419, Accessed: Apr. 04, 2023. [Online], Available: http://arxiv.org/abs/2212.00419, Dec. 2022.

[35] N. S. Agarwal, N. S. Punn and S. K. Sonbhadra, "Exploring Public Opinion Dynamics on the Verge of World War III Using Russia-Ukraine War-Tweets Dataset," KDD-UC, Washington, DC, USA, [Online], Available: https://www.kdd.org/kdd2022/papers/27_Navya Sonal Agarwal.pdf, 2022.

[36] P. Vyas, M. Reisslein, B. P. Rimal, G. Vyas, G. P. Basyal and P. Muzumdar, "Automated Classification of Societal Sentiments on Twitter with Machine Learning," IEEE Transactions on Technology and Socity, vol. 3, no. 2, pp. 100–110, DOI: 10.1109/tts.2021.3108963, 2021.

[37] R. Ibar-Alonso, R. Quiroga-García and M. Arenas-Parra, "Opinion Mining of Green Energy Sentiment: A Russia-Ukraine Conflict Analysis," Mathematics, vol. 10, no. 14, DOI: 10.3390/math10142532, 2022.

[38] B. Chen et al., "Public Opinion Dynamics in Cyberspace on Russia-Ukraine War: A Case Analysis with Chinese Weibo," IEEE Transactions on Computational Social Systems, vol. 9, no. 3, pp. 948–958, 2022.

[39] M. B. Garcia and A. Cunanan-Yabut, "Public Sentiment and Emotion Analyses of Twitter Data on the 2022 Russian Invasion of Ukraine," Proc. of the 2022 9th Int. Conf. on Information Technology, Computer and Electrical Engineering (ICITACEE 2022), pp. 242–247, DOI: 10.1109/ ICITACEE55701.2022.9924136, 2022.

[40] B. Džubur, Ž. Trojer and U. Zrimšek, "Semantic Analysis of Russo-Ukrainian War Tweet Networks," SCORES: Ljubljana, [Online], Available: http://www.scores.si/assets/papers/6258.pdf, 2022.

[41] Z. C. Steinert-Threlkeld, Twitter As Data, DOI: 10.1017/9781108529327, Cambridge Uni. Press, 2018.

[42] Mendeley Data, "Russian Aggression in Ukraine Related Tweets - Mendeley Data," DOI:10.17632/77xdt925zp.1, 2023.

[43] M. R. Baker and M. A. Akcayol, "A Novel Web Ranking Algorithm Based on Pages Multi-attribute," Int. J. of Information Technology, vol. 14, no. 2, pp. 739–749, DOI: 10.1007/s41870-021-00833-5, 2022.

[44] A. Krouska, C. Troussas and M. Virvou, "The Effect of Preprocessing Techniques on Twitter Sentiment Analysis," Proc. of the 7th Int. Conf. on Information, Intelligence, Systems and Applications (IISA 2016), pp. 1–5. DOI: 10.1109/IISA.2016.7785373, Dec. 2016.

[45] M. A. Abid, S. Ullah, M. A. Siddique, M. F. Mushtaq, W. Aljedaani and F. Rustam, "Spam SMS Filtering Based on Text Features and Supervised Machine Learning Techniques," Multimedia Tools and Applications, vol. 81, pp. 39853–39871, DOI: 10.1007/s11042-022-12991-0, 2022.

[46] K. Chen, Z. Zhang, J. Long and H. Zhang, "Turning from TF-IDF to TF-IGM for Term Weighting in Text Classification," Expert Systems with Applications, vol. 66, DOI: 10.1016/j.eswa.2016.09.009, 2016.

[47] E. Alpaydin, Introduction to Machine Learning, 4th Edn., MIT Press, DOI: 10.1007/978-3-030-74640-7_4, 2020.

[48] V. K. Vijayan, K. R. Bindu and L. Parameswaran, "A Comprehensive Study of Text Classification Algorithms," Proc. of the 2017 Int. Conf. on Advances in Computing, Communications and Informatics (ICACCI 2017), vol. 2017-Jan., pp. 1109–1113, DOI: 10.1109/ICACCI.2017.8125990, 2017.

[49] F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1–47, DOI: 10.1145/505282.505283, 2002.

[50] Y. Yang and X. Liu, "A Re-examination of Text Categorization Methods," Proc. of the 22nd Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1999), pp. 42–49, DOI: 10.1145/312624.312647, Aug. 1999.

[51] N. Jalal, A. Mehmood, G. S. Choi and I. Ashraf, "A Novel Improved Random Forest for Text Classification Using Feature Ranking and Optimal Number of Trees," J. King Saud Univ. - Comput. Inf. Sci., DOI: 10.1016/j.jksuci.2022.03.012, 2022.

[52] L. Breiman, "Bagging Predictors," Machine Learning, vol. 24, no. 2, pp. 123–140, 1996.
[53] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[54] P. Domingos, "A Few Useful Things to Know about Machine Learning," Communications of the ACM, vol. 55, no. 10, pp. 78–87, DOI: 10.1145/2347736.2347755, Oct. 2012.
[55] B. Agarwal and N. Mittal, "Text Classification Using Machine Learning Methods: A Survey," Advances in Intelligent Systems and Comp., vol. 236, pp. 701–709, DOI: 10.1007/978-81-322-1602-5_75, 2014.
[56] A. Subasi, Practical Machine Learning for Data Analysis Using Python, Elsevier, DOI: 10.1016/B978-0-12-821379-7.00008-4, 2020.
[57] H. Belyadi and A. Haghighat, Machine Learning Guide for Oil and Gas Using Python, Elsevier, DOI: 10.1016/c2019-0-03617-5, 2021.
[58] Y. Yang, "An Evaluation of Statistical Approaches to Text Categorization," Information Retrieval, vol. 1, no. 1–2, pp. 69–90, DOI: 10.1023/a:1009982220290, 1999.
[59] S. Wang and C. D. Manning, "Baselines and Bigrams: Simple, Good Sentiment and Topic Classification," Proc. of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012), vol. 2, pp. 90–94, 2012.
[60] R. Can, S. Kocaman and C. Gokceoglu, "A Comprehensive Assessment of XGBoost Algorithm for Landslide Susceptibility Mapping in the Upper Basin of Ataturk Dam, Turkey," Applied Sciences, vol. 11, no. 11, p. 4993, DOI: 10.3390/app11114993, 2021.
[61] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm," Proc. of the 13th Int. Conf. Machine Learning, pp. 148–156, DOI: 10.1.1.133.1040, 1996.
[62] W. Wang and D. Sun, "The Improved AdaBoost Algorithms for Imbalanced Data Classification," Information Sciences, vol. 563, pp. 358–374, DOI: 10.1016/j.ins.2021.03.042, Jul. 2021.
[63] A. Diera et al., "Bag-of-Words *vs.* Sequence *vs.* Graph *vs.* Hierarchy for Single- and Multi-label Text Classification," Proc. of the 60th Annual Meeting of the Association for Computational Linguistics, vol. 1: Long Papers, pp. 4038 - 4051, DOI: 10.48550/arXiv.2204.03954, 2022.
[64] A. Pinkus, "Approximation Theory of the MLP Model in Neural Networks," Acta Numerica, vol. 8, pp. 143–195, DOI: 10.1017/S0962492900002919, 1999.
[65] D. Chicco and G. Jurman, "The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation," BMC Genomics, vol. 21, no. 1, pp. 1–13, DOI: 10.1186/s12864-019-6413-7, Jan. 2020.
[66] Y. Wang, A. Sun, J. Han, Y. Liu and X. Zhu, "Sentiment Analysis by Capsules," Proc. of the World Wide Web Conference (WWW 2018), vol. 10, pp. 1165–1174, DOI: 10.1145/3178876.3186015, Apr. 2018.
[67] A. U. Rehman, A. K. Malik, B. Raza and W. Ali, "A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis," Multimedia Tools and Applications, vol. 78, no. 18, pp. 26597–26613, DOI: 10.1007/s11042-019-07788-7, Sep. 2019.
[68] S. Tam, R. Ben Said and Ö. Tanriöver, "A ConvBiLSTM Deep Learning Model-based Approach for Twitter Sentiment Classification," IEEE Access, vol. 9, pp. 41283–41293, DOI: 10.1109/ACCESS.2021.3064830, 2021.

ملخص البحث:

تبحث الدّراسة في الكيفية التي عبّر بها النّاس عن مشاعرهم على "تويتر" خلال حرْب روسيا على أوكرانيا. وحققت الدّراسة غرضين؛ فقد جمعت معلوماتٍ فريدة، كما استخدمت تعلُّم الآلة لتصنيف التّغريدات بناءً على أثرها على أحاسيس النّاس. وقد جَهِدت الدّراسة لإيجاد أكثر "الهاشتاغات" علاقةً بالحرب الرّوسية-الأوكرانية من أجل تحديد مجموعة البيانات الخاصّة بالدّراسة. كذلك عملَتْ على استخدام عددٍ من نماذج تعلُّم الآلة بغية تنظيم التّغريدات في مجموعتين.

وقد بيّنت النّتائج التّجريبية أنّ غالبية مُصنِّفات تعلُّم الآلة المستخدمة كانت ذات دقّةٍ أعلى عند استخدام مجموعات بيانات متوازنة. إلّا أنّه اتّضح أنّ استراتيجيات موازنة البيانات لم تكُنْ كلّها بالنّجاعة ذاتها. لذا كان لا بدّ من تسْليط الضّوء على مقارنة تلك الاستراتيجيات المستخدمة لموازنة البيانات في تحليل المشاعِر باستخدام تعلُّم الآلة، وذلك عبر استخدام مصنِّفات أكثر واختبارها في مدى واسعٍ من المهامّ.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

ENHANCING MEDIA STREAMING IN WIRELESS NETWORKS USING IFW-CFH ALGORITHM

Satheesh Kumar NJ and Arun CH

(Received: 11-Mar.-2023, Revised: 13-Jun.-2023, Accepted: 11-Jul.-2023)

ABSTRACT

One of the major concerns for service providers and application developers is the Quality of Experience (QoE), where high traffic congestion on the Internet leads to the degradation of video quality. However, the effectiveness of video transmission is minimized due to the network based on packet loss, bandwidth and delay. Because of bandwidth limitations, the videos transmitted are obtained in low quality. Meanwhile, various outcomes, such as reduction in throughput, re-buffering or mosaic, are determined in packet loss which validated the video streaming obtained in reliable or unreliable modes. Therefore, this paper proposes an Improved Fuzzy Weighted queueing-based Crossover Fire Hawk (IFW-CFH) algorithm for effective real-time video transmission. The objective of the IFW-CFH approach is to reduce delay, packet loss and bandwidth to enhance the video quality via two key mechanisms; namely, congestion control mechanism as well as packet scheduling mechanism. During the generation of encoded video frames, the packaged packets to the local buffer are transmitted by the scheduler using our proposed IFW-CFH algorithm. Finally, experimentation is conducted and the results show that the proposed method minimized transmission delay, packet loss and bandwidth by 13.8% for effective real-time video transmission compared to the existing methods.

KEYWORDS

Video streaming, Fire hawk, Weighted queuing, Delay, Packet loss, Crossover, Transmission.

1. INTRODUCTION

Video streaming (VS) is defined as the transferring of video files continuously from a server to an applicant. VS authorizes users to watch online videos without downloading them. Video streaming over wireless networks is mandatory for several applications and a large number of systems are used in TV shows, movies and YouTube videos [1]. Video streaming is used to reduce memory storage and bandwidth requirements when storing and converting videos [2]. Video transmission in current networks is yet limited in terms of bandwidth and suffers from packet loss, depending on the streaming of videos, re-buffering and performance degradation, where such adverse factors are very detrimental to QoE [3].

Generally, two technologies are mainly used by video content providers, which are Moving Pictures Experts Group – Dynamic Adaptive Streaming over HTTP (MPEG-DASH) and WebRTC. MPEG-DASH is the latest streaming protocol established by the MPEG as a replacement for the HLS standard; the open-source standard of HLS is designed for video- and audio-like HLS. MPEG-DASH assists adaptive-bitrate streaming and allows users to obtain good-quality videos rather than handling their networks [4]. An open-source project is WebRTC, which provides streaming with real-time latency. Some of the most common consumer-facing applications of the day use WebRTC, such as Whatsapp, Google Meet and Messenger. What makes WebRTC unique is that it depends on peer-to-peer streaming, which is a preferable resolution when streaming needs low latency [5].

Nowadays, User Datagram Protocol (UDP) is the preferred option and considers latency to deliver video frames in an unreliable mode [6]. UDP is utilized to stream audio and video over Internet Protocol (IP). The Real-Time Protocol (RTP) has been expanded to real-time video streaming of broadcast transmission. The broadcast-transmission approaches contain two transport layer protocols which are Concurrent Multipath Transfer-Stream Control Transmission Protocol (CMT-SCTP) and Multipath Transmission Control Protocol (MPTCP). These protocols provide bandwidth, increase robustness, accelerate the process completion duration and reduce the usage of multipath context [7]-[8]. Due to the diverse routing paths, packets sent *via* several paths arrive at the receiver out of order, occupying buffer resources and causing Head of Line (HOL) blocking. HOL blocking improves the performance of

Satheesh Kumar NJ(Corresponding Author) and Arun CH are with Department of Computer Science, Nesamony Memorial Christian College (Affiliated to Manonmaniam Sundaranar University), Tamilnadu, India. Emails: njsdreams@gmail.com and dassarun@gmail.com

MPTCP, because rate rollback after packet-loss identification is not caused by network traffic, where the buffer allows the applicant to obtain data from multiple servers without affecting other streams [9]-[10]. The objective of the IFW-CFH approach is to minimize delay and packet loss and reduce the bandwidth in order to enhance the video quality *via* two diverse mechanisms; namely, congestion-control mechanism as well as packet-scheduling mechanism. The vital contribution is discussed in the following points:

- A novel IFW-CFH technique is proposed for effective real-time video transmission with high video quality, minimum transmission delay and minimum packet loss.
- The convergence accuracy and the convergence speed of Fire Hawk Optimizer (FHO) are enhanced by using both horizontal and vertical crossover strategies, thereby minimizing complexity.
- Congestion-control and packet-scheduling mechanisms are employed to attain a lower bandwidth and create an effective path diversity.
- Computing and investigating the efficiency of the IFW-CFH technique are based on bandwidth, transmission delay and packet loss.

The remainder of this article is organized as follows. Section 2 describes the related works of video streaming in wireless networks by various authors. Section 3 presents the system model. Section 4 explains the proposed methodology based on video streaming. The experimental analysis is discussed in Section 5. Finally, the conclusion of the article is illustrated in Section 6.

2. LITERATUREREVIEW

To stream high-quality video through various wireless access networks, Afzal et al. [11] developed a multipath MMT-based technique. MPEG, a media-transport protocol, was used in this method to stream the video in several paths and to assist this task, a Content-Aware and Path-Aware (CAPA) technique was employed. The ns-3 DCE with various multipath networks was utilized to experiment with the functioning of the suggested CAPA and the working of CAPA was examined over wireless lossy network conditions. This technique obtained video-quality improvement in comparison with a simple scheduling strategy of Evenly Splitting and Path-Aware strategy.

Taha et al. [12] established a Quality of Experience (QoE) adaptive management system to stream video *via* wireless networks in high definition. To manage the QoE of customers and optimize assessing, a smart algorithm for the service of video streaming was suggested in this paper. The suggested algorithm contains two methodologies; namely, predicting the QoE with the machine learning technique and outperforming the previously suggested methods by enhancing the quality of the video and obtaining significant bandwidth savings.

Guo et al. [13] employed Deep Reinforcement Learning (DRL) with transcoding at the network edge for Adaptive Bit-rate Streaming (ABS) in wireless networks. In this method, communication and joint computation were suggested for Adaptive Bitrate (ABR) streaming utilizing Mobile Edge Computing (MEC) in wireless channels with time variation. To assist the ABR streaming, a combined video-quality adaptation and framework transcoding was provided using a Radio Access Network (RAN) with computing ability. An automatic DRL algorithm was created to execute the video-quality adaptation and computational-resource assignment. The suggested DRL algorithm exhibited its superiority over other current methods and it was not an omnipotent one.

To attain secured energy-efficient video streaming, Zhang et al. [14] introduced the safe Deep Q-learning Network (DQN) technique in Unmanned Aerial Vehicles (UAV)-activated wireless networks. A secured and energy-saving video streaming-activated wireless network was studied in this article, where the safe-DQN was created with the Lyapunov function to solve the issue designed as a Constrained Markov Decision Process (CMDP), which proved superiority over other current techniques.

Jiao et al. [15] developed a Dynamic Cache and Resource Allocation (DCRA) to stream the video in Orthogonal-frequency-division Multiple Access (OFDMA) by cross-tier interference to overcome problems, such as wireless resource allocation, cache placement and video-layer selection. DCRA scheme was used to overcome stochastic-optimization issues by applying the Lyapunov optimization theory. The suggested DCRA scheme proved to be more effective for streaming scalable videos and better for maintaining cross-tier interference.

Duraimurugan and Jeyarin [16] joined together and employed distributed multimedia streaming in a heterogeneous wireless network (DMSHN) to expand the QoS. QoS played a major role in distributed multimedia streaming for transferring videos from the server to the applicant. Superior quality of service was needed to produce higher-resolution videos to attain superior quality video, packet loss and reduced latency. As a result, this suggested scheme provided a superior quality of service in DMSHN by permitting the applicant to obtain data from multiple servers without affecting other streams. Meanwhile, the proposed method has not yet been exploited in direct video/audio applications of heterogeneous wireless networks.

Taha and Ali [22] adapted a smart algorithm for video streaming in wireless networks, which can be utilized in the healthcare system. This model can find out the relation between the quantization parameter (QP) and the QoS. In the rural areas as well as on the highway, a vehicular *ad-hoc* network (VANET) is utilized. So, a smart real-time multimedia traffic shaping system is illustrated by Ahmed et al. [23], which is based on distributed reinforcement learning (RMDRL). Liu and Kong [24] utilized a combination of video-streaming services and wireless networks for evaluating the performance of the video-streaming application. For this reason, an NS-3-based simulation platform is involved for the effectiveness of the system. Table 1 represents a summary of the literature review.

Table 1. Summary of literature review.

Author (s)	Technique	Uses	Pros	Cons
Afzal et al. [11]	CAPA	Streaming the video in multipath systems	High video quality	Low Performance
Taha et al. [12]	QoE	Streaming video on a wireless network	Enhanced Video Quality	High Latency
Guo et al. [13]	DRL	Wireless networks in ABS	Low Latency	Not Omnipotent
Zhang et al. [14]	Safe DQN	Activating wireless networks in UAV	Better Performance	High Costs
Jiao et al. [15]	DCRA	Overcoming the stochastic optimization	Maximizing the Time	Low Quality
Duraimurugan and Jeyarin [16]	DMSHN	Obtaining data from multiple streams	Superior-quality Video	High Packet Loss &Latency
Taha and Ali [22]	Smart algorithm	Healthcare system	Low and High Video Motions	Low Video Quality
Ahmed et al. [23]	RMDRL	Real-time multimedia traffic shaping systems	Short Frame Latency	ReducingVideo with Low Quality
Liu and Kong [24]	NS-3-based simulation platform	Performance of video-streaming application	Saving the High Costs of Real Equipment	Not to Implement in a More Realistic System

3. SYSTEM DESIGN

The significant factor for real-time video transmission is traffic control, which assures better transmission of bandwidth as well as conflicts with the other types of video flow. The congestion controller is applicable in various paths of video streaming [17]. The packets transmitted from the local buffer to the network are evaluated by congestion control. The congestion-control algorithm transmits the packets in the burst mode and these packets are queued at the median routers for generating the additional delay. The originator is responsible to send the packets in several paths to the local buffer. The bitrate of the video is adjusted by the controller to obtain an effective output [18]. The utilization function F_u is employed for equipping better service satisfaction. Generally, it is assumed to be an increasing and concave function. Here, finding of an acceptable bandwidth allocation scheme is the main goal to gain fully utilized resources. The following equation expresses the optimization problem.

$$Maximum \sum_{r \in R} T(z_r) \quad s.t \sum_{r \in R} y_r, k \leq dk \tag{1}$$

In the above equation, the user r can use the multiple routing paths for building the concurrent connections, dk indicating the bottleneck capacity, y_r, k representing the packet sending rate of path k and R signifying the number of network users.

The size and diversity of the current network system are very difficult issues in the global bandwidth allocation. The Lagrangian rate-control algorithms are to solve these issues, defined as follows:

$$\begin{aligned} L_{\arg}(Z_r,\delta) &= \sum_{r\in R} T(Z_r) + \sum_k \delta_k (d_k - \sum_{r\in R} y_r, k) \\ &= \sum_{r\in R} T(Z_r) - \sum_k \delta_k (\sum_{r\in R} y_r, k) + \sum_k \delta_k . d_k \\ &= \sum_{r\in R} T(Z_r) - \sum_{r\in R}\sum_k y_r, _k \ \delta_k + \sum_k \delta_k d_k \end{aligned} \tag{2}$$

The Kelly's shadow-price parameter is δ_k for path k. The function of the shadow-price original problem is;

$$\underset{\delta}{Minimum}\ C(\delta) \tag{3}$$

The function $C(\delta)$ is defined as:

$$C(\delta) = Maximum\ \ L_{\arg}(Z_r,\delta) + \sum_k \delta_k d_k \tag{4}$$

where,

$$L_{\arg}(\delta) = MaximumY(Z_r) - \sum_k y_r, _k \ \delta_k \tag{5}$$

The aggregate surplus is shown in Equation (7). To attain the maximum aggregate surplus while maintaining the minimum link cost, packets must be sent at a high rate. Congestion-control algorithm maintains the stability of the network system to enhance the rate of high bandwidth and bridge congestion through rate reduction. Regarding the congestion-control algorithm, an improved IFW-CFH for real-time video transmission is implemented. The main focus of the analysis is to minimize the transmission-cost liability of the packet-scheduling mode.

To discover less bandwidth,the congestion-control algorithm maximizes the rate. Then, the reduction of rate leads to link congestion and the network system's constancy is maintained. For multipath packet scheduling, the utilization theory is well employed. It is expressed in the equation below.

$$Maximum\ \ T(y) - \sum_q \delta_q y_q \tag{6}$$

In the above equation, δ_q denotes path price, q signifies path index and y_q represents packet-scheduling rate. Generating the rate of packets is y, $y_q = \beta_q y$. β_q is a splitting ratio rate, where the multipath session aggregate cost is represented by p. The expected cost of a single packet is calculated through the multipath as shown in Equation (8).

$$y_p = \sum_q y_q \delta_q \tag{7}$$

$$q = \sum_q \delta_q \beta_q \tag{8}$$

At the local buffer, the additional sent packets are queued and this happens when the packet-scheduling rate $y_q(u)$ exceeds the path-sending rate d_q.

$$\delta_q(u) = p_q(u) + p_p(u) + \int \frac{(y_q(u) - d_q(u))1(y_q(u) - d_q(u))}{d_q(u)} \tag{9}$$

The minimization of the second term is the major goal of the packet-scheduling algorithm.

$$p_u = \delta_{q,u}\gamma_{q,u} \tag{10}$$

In the above equation, $\gamma_{q,u}$ helps in scheduling the incoming packets to a particular path. The total cost of the packets scheduled with the minimum path cost at the current time should be minimized. The frame rate is considered as the sum of available bandwidth for all sub-paths $y = \sum_q d_q(u)$; *i.e.*, a single path

must not send incoming packets with the video frame rendering deadline. Additional packets that cannot be sent immediately are buffered and the path cost is increased to the least cost path shown in Equation (9). At a certain time point, the packets should be routed such that a system can achieve a balanced path cost in all terms.

4. PROPOSED METHODOLOGY

Video streaming is significant for transmitting video effectively; for effective real-time video, IFW-CFH method is proposed to enhance the performance of video-streaming quality. Two diverse mechanisms; namely, the congestion-control mechanism and the packet-scheduling mechanism, are used to transmit the video effectively as presented in the following sub-sections. Figure 1 presents the schematic flow diagram of the proposed real-time video-transmission system.

4.1 Congestion-control Mechanism

The QoE for services, such as cloud gaming, video conferencing and video streaming, is greatly impacted by congestion control, where sending packets faster can submerge the network, which leads to loss or delay in data,while sending too slow can affect the video quality. In general, the state is employed for removing the extra queues gathered at the startup stage. The state is converted to ProbeBW if the in-flight packets are below the Bandwidth Delay Product (BDP). 8 RTTs control the transferring rate by various gains [1.25, 0.75, 1, 1, 1, 1, 1, 1] at this stage and the sender could increase the transferring rate to capture the excess available bandwidth. The stage is set as ProbeRTT if the minimal RTT is not sampled in ten seconds. The RTT allows only four packets to transfer outside. For real-time video transferring, during the initial phase, the rate in the underload path rises gradually and yields a sub-optimal throughput.

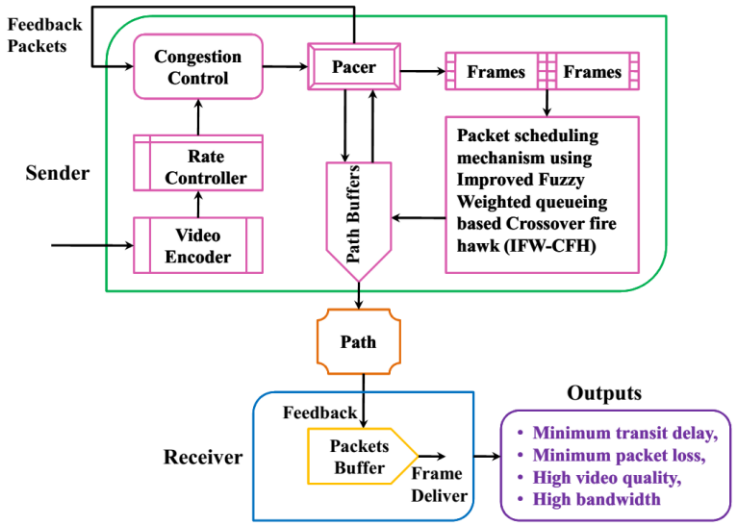

Figure 1. Real-time video-transmission system.

4.2 Packet-scheduling Mechanism

During the packet-scheduling process, the scheduler forwards the packets to the local buffer path for generating the encoded video frame with the least cost andthe path price is formulated in Equation (11).

$$H = \arg\min_{g} \lambda_g \tag{11}$$

$$\lambda_g = \frac{RTT_g(s)}{2} + \frac{O_g(s)}{f_g(s)} \tag{12}$$

$RTT_g(s)$ signifies the round-trip time. In network connection, the term $\frac{RTT_g(s)}{2}$ is defined as the utilization queuing delay and propagation delay. $f_g(s)$denotes the pacer's packet-sending cost, $O_g(s)$ and indicates the engaged buffer length at the sender.

Due to several unknown components in the routing path, the scheduling algorithm cannot guarantee sending all packets to the receiver. For example, the buffer length is occupied by routers that change paths to achieve the least arriving delay by selecting the lowest cost. After the received packets are reconstructed into video frames, they are pre-delivered to the uppermost layer.

Considering that these packets use the same frame and maybe dispatched to various paths, it is not possible for the receiver to determine whether an unfinished frame originated from late arrival or packet loss. Sent packets are cached by the sender for approximately 500 milliseconds. If a missed one is detected, it is immediately rerouted with minimal delay in transmission. The receiver chooses a maximum duration to wait for retransmitted packets. The receiver must wait to retry the miss and a complete frame is successfully carried in the keyframe. If the waiting time exceeds the maximum limit, an incomplete frame is dropped at any keyframe.

4.2.1 Fuzzy Weighted Queuing Algorithm (FWQA)

The fuzzy weighted queuing algorithm approximates packet-based Generalized Processor Sharing (GPS) regulation [19]. The backlogged flow of GPS's m^{th} flow and n^{th} flow is represented in Equation (13).

$$\frac{A_{S_m}(\gamma, d)}{A_{S_n}(\gamma, d)} \geq \frac{\omega_m}{\omega_n} \tag{13}$$

In equationabove, $A_{S_m}(\gamma, d)$ represents the number of sessions in m^{th} flow served within the range (γ, d). The number of sessions in the nth flow served within the range(γ, d) is denoted by $A_{S_n}(\gamma, d)$. In the router, each entrance of the packet is computed by using a virtual finishing time. The fuzzy weighted queuing algorithm contains different variants, such as worst-case fair weighted fair queuing (WFFQ) and self-clocked fair queuing (SCFQ). The WFFQ is corresponding to GPS and it chooses the packet for streaming without delay. However, computational complexity is a major issue at a higher speed of streaming. SCFQ is utilized to validate the beginning time of the packets, which minimizes the computational complexity. The weights of numerous classes are described in Equation (14).

$$\sum_{m=1}^{u} \omega_m = 1,\ 0.01 \leq \omega_m \leq 1 \tag{14}$$

In equation above, ω_m represents the weight of service in the m^{th} class. The total number of service classes is denoted by m. The fuzzy weighted queuing algorithm is established to stream the video efficiently and fairly. The FWQA router contains two service classes Transport Control Protocol (TCP) and UDP for video streaming. After receiving every fifty packets, the new weight queue is computed. The delay-sensitive UDP service class is assumed as an RTP carrier which is constrained to guarantee high reliability for UDP video streaming. Packet priority is achieved by computing new queue weights for the FWQA algorithm. This scenario is mathematically represented as follows.

$$S_{QL} = \frac{QL_{udp}}{QL_{tcp} + QL_{udp}} \tag{15}$$

In the equation above, S_{QL} indicates the share of queue length. QL_{tcp} is the queue length share of TCP video streaming. QL_{udp} represents the queue length of UDP video streaming.

Fuzzy Reasoning

Fuzzy reasoning is utilized to establish the fuzzy logic from the theory of fuzzy set. Reasoning should be completed by utilizing individual-based inference or combination-based inference. In this part, we deployed individual-based inference, because it is easy to implement. Fuzzy reasoning allows multiple possible truth values to be processed by a single variable. It solves the issue of pocket loss, which makes it possible to obtain consistently accurate results.

Fuzzy Control Model

The fuzzy control model is utilized to accelerate the modeling approaches for the fuzzy weighted queuing algorithm. The main goal is to minimize fall and rise times and improve and succession in UDP and TCP video streaming.

Tuning in Queue Weight Control

To identify membership functions and appropriate rules for both input and output variables to respond to the variation of video streaming, latency, re-buffering, packet loss and degradation are controlled weight values in routers. Tuning and testing are capable of maximizing the length of the rule-base operation and minimizing the computational complexity. Dividing the control variables and dynamic range of state into fuzzy membership functions are performed offline.

4.2.2 Crossover-based FireHawk Optimization

The Fire Hawk Optimization (FHO) [20] metaheuristic method imitates the fire hawk's foraging action by scattering fires as well as prey. The process of random initiation is employed to recognize the beginning position as mentioned in the following equation.

$$p_a^b(0) = p_{a,\min}^b + rand\left(p_{a,\max}^b - p_{a,\min}^b\right), \begin{cases} a=1,2,..,\mathrm{P}. \\ b=1,2,..,\mathrm{M}. \end{cases} \tag{16}$$

where P_a describes the a^{th} candidate solution from the search space area, M signifies the dimension, P denotes the total number of candidate solutions, p_a^b and $p_a^b(0)$ are the b^{th} decision variable as well as the initial position of a candidate solution, $p_{a,\max}^b$ signifies the minimum and maximum limits and $rand$ represents the random number that is uniformly distributed within the range [0,1]. T_c^d is calculated by using the following equation:

$$\mathrm{T}_c^d = \sqrt{(m_2 - m_1)^2 + (n_2 - n_1)^2}, \begin{cases} d=1,2,...,y. \\ c=1,2,...,x. \end{cases} \tag{17}$$

The distance between the d^{th} fire hawk and the c^{th} prey is T_c^d, where x and $y$indicate the total prey as well as fire hawks in the search space area. (m_1, n_1) and (m_2, n_2) describe the fire hawk and prey coordinates.

A place is considered as safe when most animals are assembled to remain sound and safe during danger. The corresponding mathematical equationsare illustrated below.

$$SP_d = \frac{\sum_{e=1}^{z} PR_e}{z}, \begin{cases} e=1,2,...,y. \\ d=1,2,...,z. \end{cases} \tag{18}$$

$$SP = \frac{\sum_{c=1}^{x} PR_c}{x}, c = 1,2,...,x. \tag{19}$$

where, PR_e is the e^{th} prey surrounded by the d^{th} fire hawk (FH_d) and PR_c is the c^{th} prey in the search space.

Crossover Strategy

This technique enhances the functioning of the firehawk, the convergence accuracy and convergence speed, which leads to the development of problems, like image multi-threshold segmentation accompanied by computational power and high complexity. Hence, this article suggested a superior firehawk algorithm with vertical as well as horizontal and crossover strategies [21].

Merging vertical, horizontal and crossover techniques with the firehawk, these techniques execute vertical, horizontal and crossover functions in every generation. After each crossover, junior individuals collide with senior individuals and junior individuals are placed higher than senior individuals for reiteration, which avoids firehawk collapse in the local optimum.

At first, the individual firehawks in the population are collected in the two-by-two non-repeating groups Qi and Qj to create the descendants by executing the horizontal crossover function in the pair of Equations (20) and (21).

$$F_{l,f}^{ke} = s_1 \cdot w_{l,f} + (1 - s_1) \cdot w_{m,f} + t_1 \cdot \left(w_{l,f} - w_{m,f}\right) \tag{20}$$

$$F_{m,f}^{ke} = s_2 \cdot w_{m,f} + (1 - s_2) \cdot w_{l,f} + t_2 \cdot \left(w_{m,f} - w_{l,f}\right) \tag{21}$$

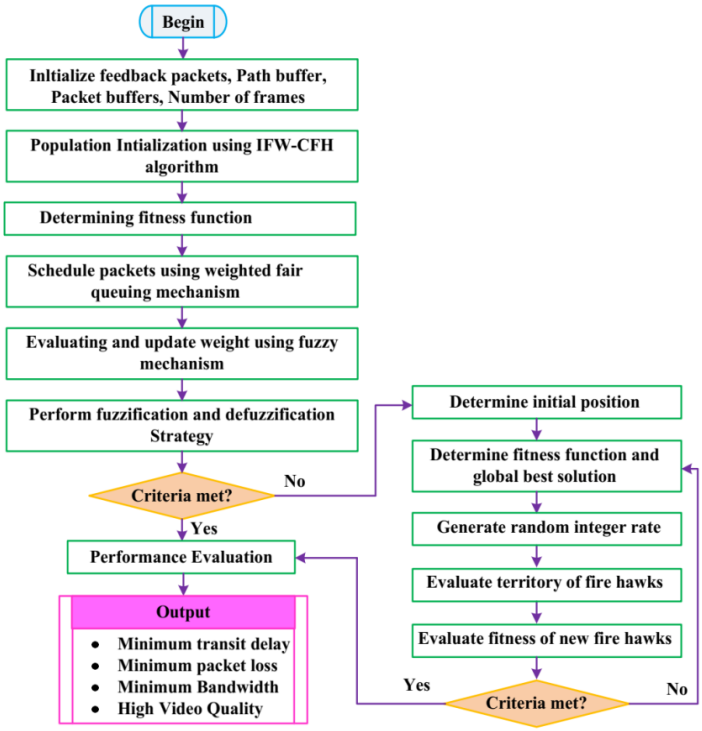

Figure 2. Flow diagram of the IFW-CFH algorithm.

where, s_1 and s_2 represent the memory coefficients. For effective packet scheduling, this paper proposes an IFW-CFH algorithm for effective real-time video transmission, which is shown in Figure 2. The diffusion coefficients denoted by t_1 and t_2 are randomly selected within the range [-1,1]. $w_{l,f}$ and $w_{m,f}$ are the dimensional vectors, where $F_{l,f}^{ke}$ and $F_{m,f}^{ke}$ are the offspring. The firehawk has an insufficiency of a robust mutation mechanism. Hence, after the completion of the horizontal crossover function performed by the algorithm, an individual population falls into the local optima to create the spatial-edge region. Due to this, there is a necessity for vertical crossover operation.

$$w_{l,f_1}^{te} = x \cdot w_{e,f_1} + (1-x) \cdot w_{e,f_2} \tag{22}$$

From the equation above, we notice a longitudinal crossover of the f_1 and f_2 dimensions of fire hawk Qi. The vertical x is an arbitrary number within the range [0, 1] and w_{l,f_1}^{te} is the individual offspring. Table 2 and Table 3 represent the Nomenclature list of all symbols and abbreviations.

Table 2. Nomenclature list of abbreviations.

Abbreviation	Description
QoE	Quality of Experience
IFW-CFH	Improved Fuzzy Weighted queueing-based Crossover Fire Hawk
VS	Video Streaming
MPEG	Moving Pictures Experts Group
DASH	Dynamic Adaptive Streaming over HTTP
UDP	User Datagram Protocol
IP	Internet Protocol
RTP	Real-time Protocol
HOL	Head of Line Blocking
FHO	Fire Hawk Optimizer
CAPA	Content-Aware and Path-Aware
DRL	Deep Reinforcement Learning
ABS	Adaptive Bit-rate Streaming

MEC	Mobile Edge Computing
DQN	Deep Q-learning Network
UAV	Unmanned Aerial Vehicles
CMDP	Constrained Markov Decision Process
DCRA	Dynamic Cache and Resource Allocation
OFDMA	Orthogonal-Frequency-Division Multiple Access
DMSHN	Distributed Multimedia Streaming in the Heterogeneous Wireless NetworkS
BDP	Bandwidth Delay Product
GPS	Generalized Processor Sharing
WFFQ	Worst-Case Fair Weighted Fair Queuing
SCFQ	Self-Clocked Fair Queuing
TCP	Transport-Control Protocol
FWQA	Fuzzy Weighted Queuing Algorithm
CMT-SCTP	Concurrent Multi-path Transfer-stream Control Transmission Protocol
MPTCP	Multipath Transmission Control Protocol
DCE	Direct Code Execution
ABR	Adaptive Bitrate
RAN	Radio Access Network

Table 3. Nomenclature list of symbols.

Symbol	Description
F_u	Utilization function
r	User can use the multiple routing paths
dk	Bottleneck capacity
y_r, k	Packet sending rate of k^{th} path
R	Number of network users
δ_q	Path price
q	Path index
y_q	Packet-scheduling rate
$RTT_g(s)$	Round-trip time
$RTT_g(s)/2$	Utilization queuing delay and propagation delay
$f_g(s)$	Pacer's packet-sending cost
$O_g(s)$	Engaged buffer length at the sender
$A_{Sm}(\gamma,d)$	The number of sessions in m^{th} flow served within the range (γ,d)
ω_m	Weight of service in m^{th} class
m	Total number of service classes
S_{QL}	Share of queue length
QL_{tcp}	Queue length share of TCP video streaming
QL_{udp}	Queue length of UDP video streaming
P_a	a^{th} candidate solution from the search space area
M and P	Dimension and total number of candidate solutions
p_a^b and $p_a^b(0)$	b^{th} decision variable as well as the initial position of a candidate solution
$p_{a,max}^b$	Signifies the minimum and maximum limits
T_c^d	Distance between the d^{th}fire hawk and the c^{th} prey
S_1 and S_2	Memory coefficients
t_1 and t_2	Diffusion coefficients
$w_{l,f}$ and $w_{m,f}$	Dimensional vectors
$F_{l,f}^{ke}$ and $F_{m,f}^{ke}$	Offspring
w_{l,f_1}^{te}	Individual offspring
x	Arbitrary number within the range [0, 1]
f_1 and f_2	Longitudinal crossover
δ_k	Shadow price of the parameter
$C(\delta)$	Dual function of the original problem

5. RESULTSAND DISCUSSION

Video streaming is performed in real-time video transmission by the IFW-CFH algorithm. For effective video transmission, the parameters link capacity, one-way propagation delay and buffer strength are used. The entire results of this paper are discussed in the following sub-sections.

5.1 Experimental Setup

The experiments of the proposed IFW-CFH algorithm are simulated on the ns-3.26 tool, which is used to validate the effectiveness of real-time video transmission. Network simulators are used to provide an accurate understanding of system behavior in communication networks that are too complex for traditional analysis methods. Practical feedback is provided to users when designing real-world systems and designers are allowed to study a system of abstraction at multiple levels. Also, a highly modular platform for wired and wireless simulations supporting various network components, protocols, traffic and routing types is presented.

5.2 Parameter Description

The parameter tuning process is conducted to effectively achieve video streaming by using the proposed IFW-CFH approach as delineated in Table 4. To define the parameters of the system parameter setting is utilized. Here, the size of the population is 30 with 100 iterations. To formulate the search space of the prey, the space is allocated by 1, 2,… etc. Also, the random interval of the horizontal crossover and the vertical crossover is defined as [-1,1] and [0,1], respectively.

Table 4. Parameter description.

Parameter	**Value**
Population size	30
Maximum number of iterations	100
Prey in search space	1,2,…etc
Random interval of horizontal crossover	[-1,1]
Random interval of vertical crossover	[0,1]

5.3 Performance Analysis

In this sub-section, the performance of one-way propagation delay, capacity estimation, packet loss, transmission delay and bandwidth are validated to attain a better achievement of the IFW-CFH method. Figure 3 illustrates the average packet transmission delay for the proposed IFW-CFH method and the existing CAPA, DRL, safe DQN and DCRA methods. It is defined as the ratio between the link length and the propagation speed over a specific medium. Here, the proposed IFW-CFH approach can obtain a lower level of transmission delay than those of the other methods. The delay is stable at 10 milliseconds.

$$PD = {}^{l}\!/_{f} \tag{23}$$

It is determined by the propagation delay and the queuing delay that demonstrate the dynamics of the employed buffer in routers. The proposed method minimized the transmission delay related to the existing methods for better efficiency and video streaming is performed accurately.

Figure 4 depicts the average rate of packet loss. The different algorithms are initiated at varied points and the packet loss rate is evaluated for CAPA, DRL, safe DQN and DCRA methods, as well as for the proposed IFW-CFH algorithm. The rate of transmission is calculated by the sender and the packet loss is stored by the receiver. By increasing the buffer, the packet loss rate is reduced and the delay in transmission is maximized in the existing methods. But, the proposed method reduced the packet loss rate compared to the state-of-the-art methods.

Figure 5 shows the average frame transmission delay. The packet-scheduling algorithms are validated in the existing methods, such as CAPA, DRL, safe DQN and DCRA, as well as the proposed IFW-CFH method. The average frame transmission delay is utilized to measure the delay packets across the Internet Protocol (IP). While simulating huge frames is created by the encoder, the proposed method attained the minimum average frame delay when compared with the existing methods.

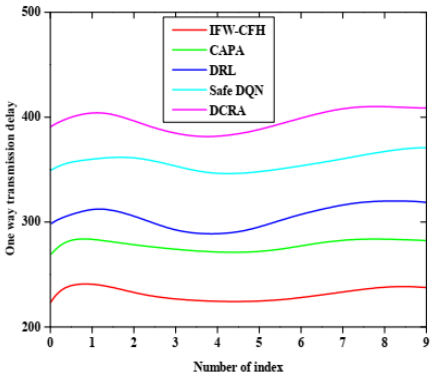

Figure 3. Comparative analysis of one-way transmission delay.

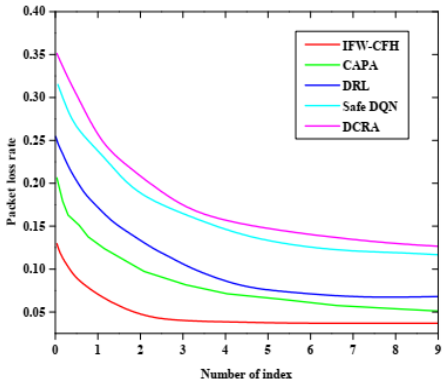

Figure 4. Comparative analysis of packet loss rate.

Table 5 delineates the comparative analysis of the proposed method with various parameters. The parameters transmission delay, packet loss rate and frame delay transmission comparison of the proposed method are performed with nine index values. Each parameter value for the proposed IFW-CFH method is minimized when compared with the existing CAPA, DRL, safe DQN and DCRA methods. The minimization of various parameter values enhances the video quality.

Figure 6 depicts the bandwidth evaluation for the existing CAPA, DRL, safe DQN and DCRA methods, as well as for the proposed IFW-CFH algorithm. The bandwidth analysis attained better efficiency in real-time video transmission. The existing methods CAPA, DRL, safe DQN and DCRA achieved 19.2%, 20.5%, 15.8% and 17.9%, while the proposed IFW-CFH algorithm achieved 13.8%, respectively. Compared to other methods, the proposed method has a lower bandwidth capability for video streaming.

Table 5. Comparative table of the proposed method with different parameters.

Number of indices	Parameters		
	Transmission delay	Packet-loss rate	Frame-transmission delay
1	223	0.032	201
2	230	0.041	180
3	218	0.064	153
4	216	0.052	148
5	217	0.040	104
6	256	0.081	176
7	239	0.078	185
8	262	0.076	182
9	213	0.075	146

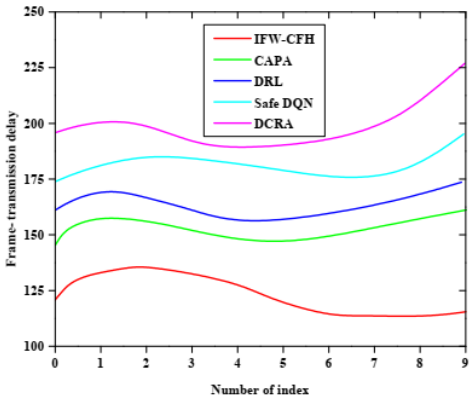

Figure 5. Comparative analysis of frame transmission delay.

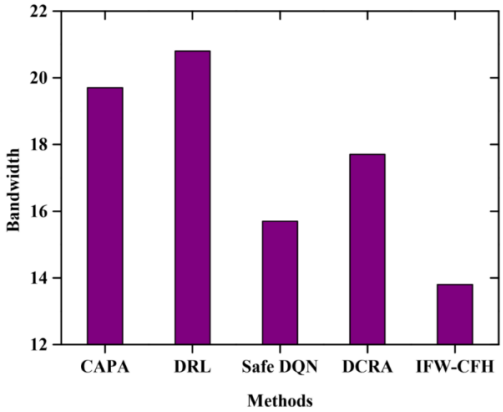

Figure 6. Comparative analysis of bandwidth with the proposed method.

Table 6 depicts the comparative analysis of the various existing methods along with the proposed method for validating the range of bandwidth values to obtain better performance of video quality. CAPA, DRL, Safe DQN and DCRA are the existing methods that obtained bandwidth value ranges of

19.2%, 20.5%,15.8% and 17.9%, respectively, whereas the proposed IFW-CFH method attained 13.8%. This reduction of bandwidth in the proposed method will improve the quality of video when compared with the existing methods.

Table 6. Comparative analysis of bandwidth for the proposed method and the existing methods.

Existing Methods	Bandwidth (%)
CAPA	19.2
DRL	20.5
Safe DQN	15.8
DCRA	17.9
Proposed IFW-CFH	13.8

6. CONCLUSION

In this article, an IFW-CFH algorithm is proposed to enhance video quality. Different parameters are used to estimate the efficiency of video transmission. The QoE is necessary for generators and service providers for performing video streaming. The reduction in throughput, re-buffering or mosaic is determined in packet loss to validate whether video streaming is employed in a reliable or unreliable mode. The proposed approach is validated by various measures, such as propagation delay, bandwidth and packet loss. Congestion control and packet scheduling are the two crucial objectives that are used to handle the lower bandwidth as well as to generate effective path diversity. In a comparative analysis, the proposed IFW-CFH algorithm is compared with CAPA, DRL, safe DQN and DCRA methods to validate the performance of video quality. The IFW-CFH method attained effective real-time video transmission by reducing the bandwidth by 13.8%, as well as transmission delay and packet loss when compared to the existing methods CAPA, DRL, safe DQN and DCRA of 7.8%, 11.9%, 9.7% and 5.2%, respectively. In the future, the proposed approach will be used to increase the efficiency of video transmission by implementing a new paradigm based on the privacy criterion and cost of providing the desired video quality of tele-training videos. IFW-CFH algorithm should be implemented in a real system, like WebRTC and SignalR and optimal settings should be found by the adjusting parameters to change video quality against bandwidth.

REFERENCES

[1] T. A. Q. Pham, K. D. Singh, J. A. Rodríguez-Aguilar et al., "AD3-GLaM: A Cooperative Distributed QoE-based Approach for SVC Video Streaming over Wireless Mesh Networks," Ad Hoc Networks, vol. 80, no. 2018, pp. 1-15, 2018.

[2] Y. Yu and S. Lee, "Remote Driving Control with Real-time Video Streaming over Wireless Networks: Design and Evaluation," IEEE Access, vol. 10, pp. 64920-64932, 2022.

[3] J. Du, F. R. Yu, G. Lu, J. Wang, J. Jiang and X. Chu, "MEC-assisted Immersive VR Video Streaming over Terahertz Wireless Networks: A Deep Reinforcement Learning Approach," IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9517-9529, 2020.

[4] J. S. Leu and C. W. Tsai, "Practical Design of a Proxy Agent to Facilitate Adaptive Video Streaming Service Across Wired/wireless Networks," Journal of Systems and Software, vol. 82, no. 11, pp. 1916-1925, 2019.

[5] G. Bijur, R. Mundugar, V. Mantooret al., "Estimation of Adaptation Parameters for Dynamic Video Adaptation in Wireless Network Using Experimental Method," Computers, vol. 10, no. 4, p. 39, 2021.

[6] Y. S. Baguda, "Energy-efficient Biocooperative Video-aware QoS-based Multiobjective Cross-layer Optimization for Wireless Networks," IEEE Access, vol. 8, pp. 127034-127047, 2020.

[7] R. Murugadoss and M. M. V. M. Kumar, "The Quality of Experience Framework for HTTP Adaptive Streaming Algorithm in Video Streaming over Wireless Networks," Int. J. Fut. Gener. Commun. Netw., vol. 13, pp. 1491-1502, 2020.

[8] M. Morshedi and J. Noll, "Estimating PQoS of Video Streaming on Wi-Fi Networks Using Machine Learning," Sensors, vol. 21, no. 2, p. 621, 2021.

[9] S. Kumarganesh, S. Anthoniraj, T. S. Kumar et al., "A Novel Analytical Framework is Developed for Wireless Heterogeneous Networks for Video Streaming Applications," Journal of Mathematics, vol. 2022, Article ID: 2100883, 2022.

[10] S. Felici-Castell, M. García-Pineda, J. Segura-Garcia, R. Fayos-Jordan and J. Lopez-Ballester, "Adaptive Live Video Streaming on Low-cost Wireless Multihop Networks for Road Traffic Surveillance in Smart Cities," Future Generation Computer Systems, vol. 115, pp. 741-755, 2021.

[11] S. Afzal, C. E. Rothenberg, V. Testoni, P. Kolan and I. Bouazizi, "Multipath MMT-based Approach for Streaming High Quality Video over Multiple Wireless Access Networks," Computer Networks, vol. 185, p. 107638, 2021.
[12] M. Taha, A. Canovas, J. Lloret and A. Ali, "A QoE Adaptive Management System for High Definition Video Streaming over Wireless Networks," Telecommunication Systems, vol. 77, pp. 63-81, 2021.
[13] Y. Guo, F. R. Yu, J. An, K. Yang, C. Yu and V. C. Leung, "Adaptive Bitrate Streaming in Wireless Networks with Transcoding at Network Edge Using Deep Reinforcement Learning," IEEE Transactions on Vehicular Technology, vol. 69, no. 4, pp. 3879-3892, 2020.
[14] Z. Zhang, Q. Zhang, J. Miao, F. R. Yu, F. Fu, J. Du and T. Wu, "Energy-efficient Secure Video Streaming in UAV-enabled Wireless Networks: A Safe-DQN Approach," IEEE Transactions on Green Communications and Networking, vol. 5, no. 4, pp. 1892-1905, 2021.
[15] L. Jiao, H. Yin and Y. Wu, "Dynamic Resource Allocation for Scalable Video Streaming in OFDMA Wireless Networks," IEEE Access, vol. 8, pp. 33489-33499, 2020.
[16] S. Duraimurugan and P. J. Jayarin, "Maximizing the Quality of Service in Distributed Multimedia Streaming in Heterogeneous Wireless Network," Multimedia Tools and Applications, vol. 79, no. 5-6, pp. 4185-4198, 2020.
[17] W. Liu, H. Zhang, H. Ding and D. Yuan, "Delay and Energy Minimization for Adaptive Video Streaming: A Joint Edge Caching, Computing and Power Allocation Approach," IEEE Transactions on Vehicular Technology, vol. 71, no. 9, pp. 9602-9612, 2022.
[18] S. Zhang, W. Lei, W. Zhang, Y. Guan and H. Li, "Congestion Control and Packet Scheduling for Multipath Real Time Video Streaming," IEEE Access, vol. 7, pp. 59758-59770, 2019.
[19] S. Nandhini, "Low Latency Weighted Fair Queuing for Real time Flows with Differential Packet Dropping," Indian Journal of Science and Technology, vol. 8, no. 22, pp. 1-8, 2015.
[20] M. Azizi, S. Talatahari and A. H. Gandomi, "Fire Hawk Optimizer: A Novel Metaheuristic Algorithm," Artificial Intelligence Review, vol. 56, pp. 287–363, 2022
[21] G. Ma and X. Yue, "An Improved Whale Optimization Algorithm Based on Multilevel Threshold Image Segmentation Using the Otsu Method," Engineering Applications of Artificial Intelligence, vol. 113, p. 104960, 2022.
[22] M. Taha and A. Ali, "Smart Algorithm in Wireless Networks for Video StreamingBased on Adaptive Quantization," Concurrency and Computation: Practice and Experience, vol. 35, no. 9, p. e7633, 2023.
[23] A. A. Ahmed, S. J. Malebary, W. Ali and O. M. Barukab, "Smart Traffic Shaping Based on Distributed Reinforcement Learning for Multimedia Streaming over 5G-VANET Communication Technology," Mathematics, vol. 11, no. 3, p. 700, 2023.
[24] G. Liu and L. Kong, "Simulation of Video Streaming over Wireless Networks with NS-3," arXiv preprint, arXiv:2302.14196, 2023.

ملخص البحث:

إنّ إحدى المشكلات التي تواجه مقدِّمي الخدمة ومطوّري التّطبيقات تتمثّل في "جودة الخبرة" (QoE)، حيث يؤدّي "الازدحام المروري" على الإنترنت الى تدهور جودة صور الفيديو. وتقلّ فعالية نقْل صور الفيديو في الشبكات لأسبابٍ تتعلّق بفقْد الحُزَم وعرض النّطاق والتّأخير. وبسبب محدِّدات عرض النّطاق، فإنّ صور الفيديو المنقولة تكون ذات جودة منخفضة.

لذا فإنّ هذه الورقة البحثية تقترح خوارزمية من نوع (IFW-CFH) لنقْل صور الفيديو بفعالية في الزّمن الحقيقي. وتهدف الخوارزمية المقترّحة الى تقليل التّأخير وفقْد الحُزَم وعرض النّطاق من أجْل تحسين جودة صور الفيديو المنقولة من خلال آليّتين هما: آليّة التّحكم بالازدحام، وآليّة جدولة الحُزَم.

وقد أشارت نتائج التّجريب الى أنّ الطّريقة المقترحة قلّلت من تأخير النّقل وفقْد الحُزَم وعرض النّطاق بنسبة 13.8%، وهي تفوّقت بذلك على تقنياتٍ أخرى مستخدَمة في دراسات سابقة للحصول على نقْلٍ فعّال لصور الفيديو في الزّمن الحقيقي.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

MOBILE U-NET V3 AND BILSTM: PREDICTING STOCK MARKET PRICES BASED ON DEEP LEARNING APPROACHES

D. Murahari Reddy and R. Balamanigandan

(Received: 24-Apr.-2023, Revised: 21-Jun.-2023, Accepted: 18-Jul.-2023)

ABSTRACT

Stock-market prediction is the task of forecasting future movements or trends in stock prices or overall market behavior. Investors can able to locate companies that offer the highest dividend yields and lower their investment risks by using a trading strategy. It's important to note that predicting stock markets accurately is extremely challenging and no approach can guarantee consistent success. Markets are influenced by a multitude of factors and there is inherent uncertainty involved. For instance, predicting stock-market prices is commonly used in financial disciplines, such as trade-execution strategies, portfolio optimization and stock-market forecasting. Therefore, it's crucial to approach stock-market prediction cautionsly and use it as a tool for informed decision-making rather than relying solely on predictions. To overcome the challenges, we proposed a new hybrid deep-learning technique to forecast future stock prices. Deep learning has recently enjoyed considerable success in some domains due to its exceptional capacity for handling data. In this research, we propose a hybrid technique of Mobile U-Net V3 and BiLSTM (Bi-Long Short-Term Memory) to predict stock prices. Initially, we utilize the min-max normalization method to normalize the input data in the preprocessing stage. After normalizing the data, we utilize hybrid deep learning techniques of Mobile U-Net V3 and BiLSTM to predict the closing price from stock data. To experiment, we collect data from Apple, Inc. and S&P 500 stock. The evaluation metrics Pearson's Correlation (R), Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Normalization Root Mean Squared Error (NRMSE) were utilized to calculate the outcomes of the DL stock-prediction methods. The Mobile U-Net V3-BiLSTM model outperformed other techniques in forecasting stock-market prices.

KEYWORDS

Stock market, Prediction, Future stock prices, Artificial intelligence, Deep learning, Mobile U-Net V3, BiLSTM.

1. INTRODUCTION

The stock market is an important phase of every country's economy. It is one of the most significant investment opportunities for investors and businesses. Through an Initial Public Offering, a firm can make a sizable profit by growing its enterprise. The company's shareholder bonus program provides dividends at an opportune moment for investors to buy additional equities and profit from them [1]-[3]. An investor who trades on the stock market can do so with stocks.

For accurate decision-making to hold, sell or purchase extra stocks, stock traders must predict patterns in the stock-market behavior. Stock traders must purchase stocks the prices of which are anticipated to rise soon and sell those the prices of which are anticipated to fall to make a profit. Stock traders can make substantial profits if they correctly forecast stock-price patterns [4]-[6]. Consequently, forecasting future stock-market patterns is crucial for stock traders' decision-making. So far, stock markets are most challenging to forecast and unpredictable and external influences, such as daily financial news and social media, have an immediate negative or positive impact on stock values. For a precise stock-market prediction, these aspects must be taken into account.

Although investing in the stock market carries some risk, it is one of the most effective ways to make sizable returns when carried out with discipline [7]-[8]. Investors, however, cannot fully evaluate such a vast volume of financial news and social-media information. Investors must therefore use an automated decision-support system, since it will automatically assess stock movements using such massive amounts of information. In earlier studies on stock prediction, machine-learning algorithms were employed to forecast the stock market utilizing historical data. Various predictive models that employ either type of data have been suggested. Investors can utilize the information from these systems

D. Murahari Reddy and R. Balamanigandan are with Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, TamilNadu, India. Emails: dmuraharireddy@gmail.com and drrbalamanigandan@gmail.com

to help them decide whether to sell or buy a stock [9]-[12]. Nevertheless, using only one type of data could not result in improved stock-market prediction accuracy.

In a technical analysis method, historical data has been utilized to examine information to forecast future stock-market patterns. Researchers analyzed historical stock-price data using a various of machine-learning methods, including DL and regression analysis. However, it is crucial to consider external factors, because unexpected events that are discussed on social media and in the news can also an impact on stock prices. Those who want to invest in the stock market sometimes have no idea how the market operates [13]-[14]. They are unable to optimize their gains, because they are unsure of which shares to buy and which to sell. These investors are aware of that connected news influence the stock-market growth. They must therefore have timely and reliable information regarding stock-market listings to make informed trading selections. As financial news on websites are a reliable source of this information, the majority of these websites have developed into important informational resources for traders. Expectations of investors based on financial information as a trading technique, however, could not be sufficient [15]. In this research, we utilized novel DL techniques to predict the stock-market analysis. Here, we utilize the min-max normalization approach to normalize the given input data in the preprocessing stage and then predict the stock-market analysis. To predict the stock-market analysis of S&P 500 stock data and Apple, Inc. stock data, we employed hybrid Mobile U-Net V3 and BiLSTM techniques. This model analyzes the forecast of the closing prices of these two companies. The key contribution is as follows:

- Predicting the stock-market analysis more closely is a trending domain and many individuals using stock-market trades in recent years.
- In the preprocessing stage, the input data is normalized using the min-max normalization method.
- To predict the stock-market analysis, we utilized hybrid Mobile U-Net V3 and BiLSTM techniques to determine whether a stock will go up or down.
- For the experiments, we used two datasets; namely, S&P 500 stock data and Apple, Inc. data.

The remaining sections of the research are divided into the following steps, Section 2 lists the literature that is related to the paper. The problem statement is given in Section 3. The proposed technique is explained in Section 4. Section 5 presents the outcomes. Finally, Section 6 presents the conclusions.

2. LITERATURE SURVEY

Many research studies are recently focused on forecasting the prices of stock markets and currency exchanges. The study "Framework for predicting and modeling stock-market prices based on deep-learning algorithms" was conducted by the authors in [16]. They suggested a design based on a hybrid of CNN-LSTM to forecast the closing prices of Apple and Tesla companies. These forecasts were produced utilizing information gathered during the previous two years. The outcomes of the DL stock-estimation techniques were computed using the RMSE, MSE, R and NRMSE measures.

The study "Stock-price prediction based on deep neural networks" was conducted by the authors in [17]. Financial product price information is viewed as a 1-D series produced by the projection of a chaotic model made up of numerous components into the time dimension and the price series is rebuilt utilizing the time-series phase-space reconstruction (PSR) approach. To predict stock prices, a deep neural network-based prediction algorithm is developed based on the PSR technique using LSTMs for DL. Several stock indices for various periods are predicted using the suggested prediction model, as well as some other methods.

The study "A CNN-BiLSTM-AM method for stock-price prediction" was conducted by the authors in [18]. To forecast the stock closing price of the following day, they presented a CNN-BiLSTM-AM method. CNN, BiLSTM and AM make up this technique. To retrieve the attributes from the input information, a convolutional neural network is utilized. The stock closing price of the following day is predicted by BiLSTM using the retrieved attribute data. To increase forecast accuracy, AM is utilized to capture how feature states affected the closing price of the stock at various points in the past. This technique and seven other techniques are utilized to forecast the Shanghai Composite Index's stock closing price for 1000 trading days to demonstrate the method's efficacy.

The study "An innovative neural network approach for stock-market prediction" was conducted by the

authors in [19]. To predict the stock market, they presented an automated encoder with LSTM neural network and the embedded layer in a deep LSTM neural network. To vectorize the data in these two approaches so that an LSTM neural network can forecast the stock, they employed the embedding layer and the automatic encoder, respectively.

The study "An improved deep-learning model for predicting stock-market price time series" was conducted by the authors in [20]. In contrast to conventional models, they suggested an approach for projecting stock closing prices that provides a more accurate prediction. The components that form this deep hybrid framework are the deep-learning predictor portion, the data processing portion and the predictor optimization algorithm. Data-processing techniques include preparation based on the empirical wavelet transform (EWT) and post-processing based on the outlier robust extreme learning machine (ORELM) approach. The major component of the mixed frame, an LSTM network-based DL network predictor, is jointly improved by the dropout approach and PSO algorithm. Table 1 shows the literature survey's comparison.

Table 1. The comparison table of literature survey.

Ref.	Technique	Dataset	Advantages	Disadvantages
[16]	CNN-LSTM	Tesla and Apple Stock-market Data	The majority of economies and people have long sought reliable future predictions, which may be provided by this method.	The authors didn't make use of the sentiment data that came from the analysis of the financial markets.
[17]	LSTM	S&P 500	Predicting and analyzing financial data nonlinear and accurately achieving better accuracy.	The running time complexity was higher than in other techniques.
[18]	CNN-BiLSTM-AM	Shanghai Composite Index Stock	It can serve as a useful resource for investors looking to maximize the return on their investments and can also be used by those conducting research on financial time-series information to gain first-hand experience.	To enhance the accuracy of the outcomes, the parameters of the model were not primarily changed.
[19]	ELSTM	Shanghai A-Share Composite Index	The Shanghai A-Share Composite Index can be predicted more accurately using the applied methods.	Although the algorithms can somewhat enhance the effect of the Shanghai A-share composite index, there are still certain issues with historical-information input. The stock market underutilizes textual data such as news.
[20]	LSTM	S&P 500	The model that incorporates decomposition and error correction makes predictions more accurately. The dropout technique and PSO algorithm can raise the LSTM network's forecasting precision.	The time complexity is higher.

3. PROBLEM STATEMENT

Most of the time, financial analysts who invest in the stock market lack awareness of market behavior. They have a problem with trading, since they are unable to decide which stocks to sell or buy to increase their profits. The stock market's entire knowledge base is readily available in the modern world. It would be quite challenging to analyze all of this data manually or individually. It is therefore necessary to automate the process. Data-mining methods are handy in this situation. Stock-price changes are usually complex. It has always been important for traders to predict changes in stock prices. Shareholders' investment challenge can be greatly decreased by making a realistic and precise estimate of the changes in stock prices.

3.1 Aim of This Research

To overcome the above problems, we utilized novel DL techniques to predict the stock-market analysis. Here, we utilize the min-max normalization approach to normalize the given input data in preprocessing stage. After that, we predict the stock-market analysis. To forecast the stock-market analysis of S&P 500 stock data and Apple, Inc. stock data, we employed hybrid Mobile U-Net V3 and BiLSTM techniques. This model analyzes the forecast of the closing prices of these two companies.

4. PROPOSED METHODOLOGY

On the stock market, stocks can be sold, swapped and moved about. Enterprises have used this to raise finance for 400 years. By problem stocks, a sizeable number of capitals is introduced to the stock market. In this research, we utilized new DL techniques to predict the stock-market analysis. Here, we utilized the min-max normalization approach to normalize the given input data in the preprocessing stage. After that, we predict the stock-market analysis. To predict the stock market analysis from S&P 500 stock data and Apple, Inc. stock data, we employed hybrid Mobile U-Net V3 and BiLSTM techniques. This model analyzes the forecast of the closing prices of these two companies. Figure 1 shows the architecture of the proposed methodology.

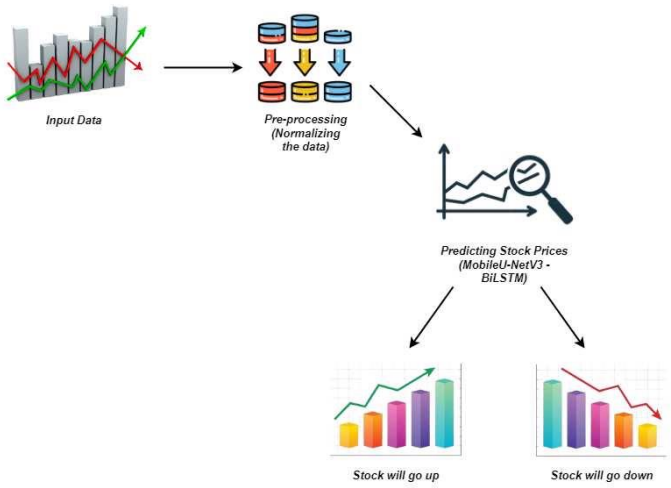

Figure1. The proposed methodology architecture of stock-market prediction.

4.1 Preprocessing

The historical data, such as opening price, high and low price, closing price and volume of these variables, can be used as input features to train a Mobile U-Net V3-BiLSTM model for predicting closing prices. The input data is normalized or scaled to a small range, such as between 0 and 1, to ensure stable training. In the preprocessing stage, we normalize the input data using the min-max normalization method. When using a large amount of stock-price data, normalization is a helpful method for scaling the stock information so that it fits within a specific range. Gradient descent is accelerated and becomes more precise after normalization [21]. By applying a linear change to the starting date, scaling the data between specific ranges is typically done using min-max normalization. The notations $x_{\min}$ $x_{\max}$, respectively, stand for an attribute's lowest and highest values. To calculate the distinction between the two values, the value in the range [$x_{\min}$ - $x_{\max}$] is used for the computation of the value x. The normalized data for S&P 500 and Apple is shown in Figure 2:

$$z_n = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \left(New_{\max_x} - New_{\min_x} \right) + New_{\min_x} \tag{1}$$

where the variables $x_{\min}$ and $x_{\max}$ stand for the lowest and highest values, respectively. The notation $New_{\min_x}$ represents the smallest integer, whereas the notation $New_{\max_x}$ represents the largest integer.

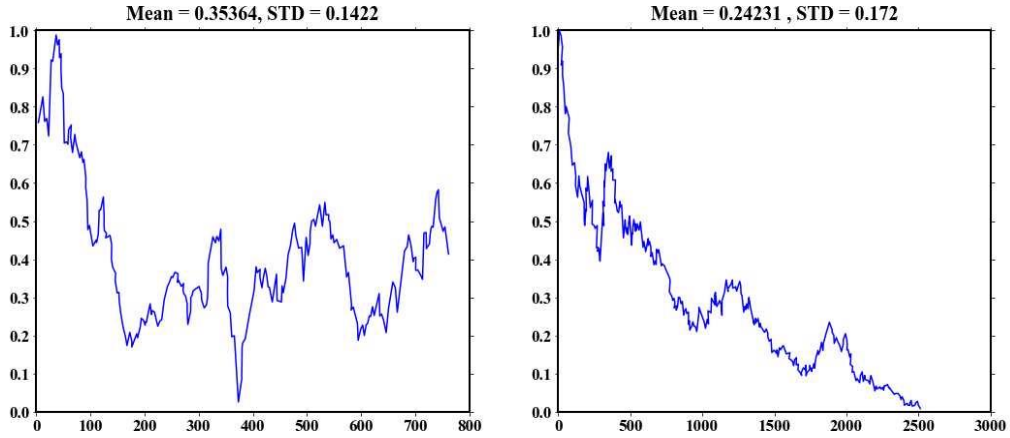

Figure 2. The normalization data of (a) S&P 500, (b) Apple.

4.2 Prediction Models

In the prediction stage, we utilize hybrid deep-learning techniques of Mobile U-Net V3 and BiLSTM. These methods are predicting the closing price of given two datasets' stock data. Combining these two algorithms may produce a better output compared to other algorithms.

4.2.1 BiLSTM

4.2.1.1 Long Short-term Memory Network (LSTM)

J. J. Hopfield suggested a Recurrent Neural Network (RNN) in 1982 for handling sequence data. The outcome of an RNN is linked back to the input by feedback, acting as a dynamic memory, unlike a standard ANN. For short-term predicting, this network showed the best performance, but when it comes to long-term predicting, it becomes unreliable. This instability is caused by the gradient explosure or by sudden, significant changes in training weights. By enabling memory cells in the hidden layer (s), the LSTM network solve the gradient-explosure problem. Figure 3 depicts the LSTM's basic architecture.

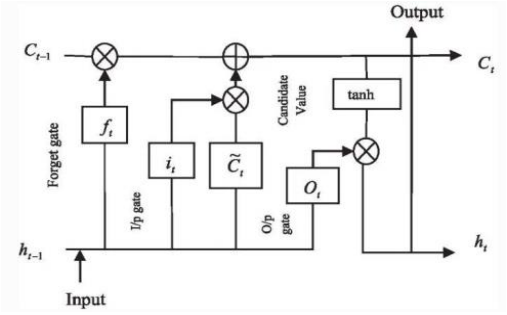

Figure 3. The basic architecture of LSTM.

Every LSTM cell has an input gate (i_t), a forget gate (f_t) and an output gate (O_t) that can accept or reject data. The network has ignored the preceding cell state "C_{t-1}" for a forward-movement function. Inputs '$GHI_i(t)$', 'h_{t-1}' and 'b_f' of the forget-gate bias are the three inputs that the LSTM network currently has at the time't'. Hence, the activation values can be written as follows:

$$f_t = \sigma\left(W_f \cdot \left[h_{t-1}, GHI_i(t)\right] + b_f\right) \tag{2}$$

The network uses the following equations to determine whether the data needs to be destroyed or maintained.

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, GHI_i(t)\right] + b_i\right) \tag{3}$$

$$\tilde{C}_t = \tanh\left(W_C \cdot \left[h_{t-1}, GHI_i(t)\right] + b_C\right) \tag{4}$$

$$C_t = f_t * C_{t-1} + i_i * \tilde{C}_t \tag{5}$$

The memory cell's outcome now changes to:

$$O_t = \sigma\left(W_O \cdot \left[h_{t-1}, GHI_i(t)\right] + b_O\right) \tag{6}$$

$$h_t = O_t * \tanh\left(C_t\right) \tag{7}$$

where "b_i", "b_f", "b_c" & "b_o" are bias vectors of the LSTM network; σ is a sigmoid function ranging from '0' to '1' and "W_i", "W_f", "W_c" & "W_o" are weight vectors of the LSTM network.

4.2.1.2 Bidirectional Long Short-term Memories (BiLSTM)

The BiLSTM technique is employed to predict the stock prices of S&P 500 and Apple data combined with the Mobile U-Net V3 technique. The forward and backward LSTM networks that constitute the BiLSTM network allow for both forward and backward data processing [22]. The data's underlying patterns and attributes are captured *via* processing in the reverse direction, which LSTM often ignores. Fig. 4 depicts the basic design of the BiLSTM network.

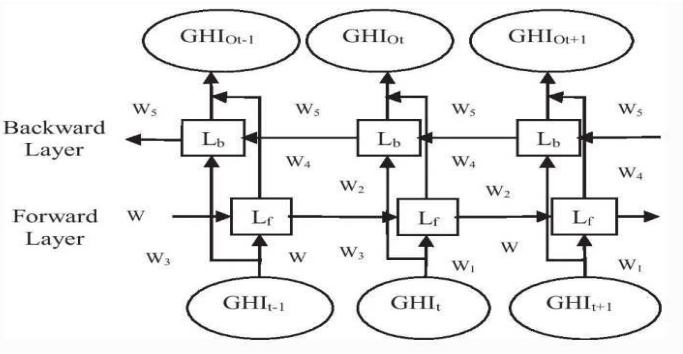

Figure 4. The basic architecture of the BiLSTM technique.

The network was updated using the output sequence "$GHI_o(t)$" and the forward hidden layer "L_f", as well as the backward hidden layer "L_b". The network iteratively updates from "T" to "1" and from "1" to "T" in backward and forward directions, respectively. The network's updated parameters can be stated mathematically as follows:

$$L_f = \sigma\left(W_1 GHI_i(t) + W_2 L_{f-1} + b_{L_f}\right) \tag{8}$$

$$L_b = \sigma\left(W_3 GHI_i(t) + W_5 L_{b-1} + b_{L_b}\right) \tag{9}$$

$$GHI_o = W_4 L_f + W_6 L + b_{GHI_o} \tag{10}$$

where L_b is backward pass, $GHI_o(t)$ stands for the final output layers and L_f is the forward pass. 'W' is the weight coefficient and 'b_{L_f}', 'b_{L_f}'& 'b_{GHI_o}' are the biases. Bidirectional LSTM (BiLSTM) is a variant of the RNN structure that incorporates data from both previous and future contexts to make predictions or analyze sequential data. BiLSTM addresses the limitation of traditional LSTMs by capturing dependencies in both backward and forward directions.

Advantages: BiLSTM networks have the advantage of being able to capture long-term dependencies in the input sequence, which is important for many NLP tasks. However, they can be computationally expensive and may require a much to train effectively.

Disadvantages: BiLSTM networks are powerful, but computationally expensive. BiLSTM is a much slower model and requires more time for training.

4.2.2 The Proposed MobileU-NetV3 Method

The Mobile U-Net V3 technique is employed with BiLSTM for predicting the S&P 500 and Apple stock prices. To make use of MobileNetV3's powerful predicting capabilities for stock-price data, the

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

proposed model contains the U-Net structure with it, giving it the designation MobileU-NetV3. It is a lightweight deep neural network constructed with depth-wise convolution. Mobile U-Net V3 is a variant of the U-Net architecture that is designed for efficient and accurate prediction tasks, particularly in scenarios with limited computational resources, such as mobile devices. The proposed MobileUNetV3 model's structure comprises all of its building pieces and the input feature maps [23]. In the encoder section, down-sampling is used in conjunction with the chosen MobileNetV3 layers to minimize the data size.

The system uses MobileNet V3 as the backbone encoder of the U-Net structure and passes the input data *via* it (16, 20, 38, 93 and 214). For example, layer 16 modifies the data size with 64 bands to 112 × 112. Layer 20 modifies the data size with 64 bands to 56 × 56. Layer 38 modifies the data size with 78 bands to 28 × 28. Layer 93 modifies the data size with 240 bands to 14 × 14. Finally, layer 214 modifies the data size with 960 bands to 7 × 7. After that, each layer of MobileNetV3 is concatenated with the preceding output layer and up-sampled using the U-Net decoder. The output Layer 93 is concatenated with layer 214, the first up-sampling, which has a data size of 14 by 14 and 512 bands. To obtain the output, a transposed convolution layer, which is another name for a de-convolution layer, is utilized along with a Softmax activation method.

To efficiently train the Mobile U-Net V3 framework, indicated by L_{ce}, the L_{ce} training set's loss is calculated as follows:

$$L_{ce} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C} 1\left(y_i = k\right)\ln\left[\, p\left(y = k \middle| x\right)\right] \tag{11}$$

where C is the amount of classes and N is the amount of training samples.

Mobile U-Net V3 has a U-Net-like structure and they have some similarities. There are some variations, though. The connecting way in the U-Net framework is easier. Deconvolution is used in the U-Net model's expanding or up-sampling phase to decrease the amount of feature maps while enhancing their dimensions. To preserve the pattern structure in the data, feature mappings from the network's contracting portion are replicated in the expanding portion. The MobileNetV3 structure, on the other hand, is optimized utilizing an algorithmic search technique to determine the optimal structure and MobileUNetV3 makes use of a more advanced contracting component based on that architecture. The Mobile U-Net V3-BiLSTM technique is utilized to pretrain parameters, such as dropout rate, dropout period and batch size, in order to improve the overall accuracy.

Advantages: Mobile U-Net V3 maintains the right ratio of precision and model size. It adopts various techniques, such as skip connections and multi-scale feature fusion, to capture both local and global information in the input data. This helps improve the accuracy of tasks.

Disadvantages: The performance of Mobile U-Net V3 can vary depending on the specific dataset and domain characteristics. Since the model is trained on a particular set of data, its effectiveness may be limited when applied to domains or datasets that significantly differ from the training data.

5. RESULTS AND DISCUSSION

The first half of this section uses the dataset analysis to anticipate stock-market prices and comparing our method with "state-of-the-art" approaches.

5.1 Experimental Setup

An Intel i5 2.60 GHz processor and 4 GB of RAM power Windows 10 on this device. Python, KERAS and TensorFlow are used to perform the investigations against the backdrop of the Anaconda3 environment. The Apple, Inc. dataset [24] and the S&P 500 stock dataset [25] are used in this study as validation datasets to determine the effectiveness of our proposed approach. In Table 2, the test environment is displayed.

Table 2. Test environment.

Project	Environment
System	Python
Processor	Intel i5 2.60 GHz
RAM	4GB

5.2 Dataset Description

5.2.1 Apple, Inc. Dataset [24]

Apple, Inc. is a multinational innovation company that designs, produces and advertises a various of electronic goods, such as laptops, tablets, smartphones, wearable technologies and accessories. The New York Stock Exchange's ticker symbol for the company's stock is AAPL. Among the Company's valuable products, product lines include the iPhone, the Mac line of desktop and laptop computers, the Watch, the TV and the iPad. The rapidly expanding service division of the business, as well as its extra revenue sources, are exemplified by the Apple's digital streaming entertainment services and iCloud cloud service, including Apple TV+ and Apple Music. This dataset provides historical statistics on Apple, Inc.'s share prices. Every day, one can get information on the Company's share prices.

5.2.2 S&P 500 Stock Dataset [25]

Data for six stock indices, including the S&P 500, Nikkei 225 (N 225), Dow Jones industrial average (DJIA), Hang Seng Index (HSI), China Securities Index 300 (CSI 300) and ChiNext index, was gathered from relevant organizations, as well as TuShare financial data interface (tushare.org) and Yahoo Finance (finance.yahoo.com). The prices at which trading day's markets closed used as the analytical data. Table 3 shows the training, testing and validation values for each dataset.

Table 3. Training, testing and validation values for each dataset.

Dataset	Training	Testing	Validation
Apple, Inc. dataset	65%	20%	15%
S&P 500 stock dataset	65%	20%	15%

5.3 Evaluation Metrics

The approaches used to test the forecasting effectiveness of the Mobile U-Net V3-BiLSTM and BiLSTM approaches used the RMSE, MSE, R and NRMSE metrics:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2 \tag{12}$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{\left(y_{i,\exp} - y_{i,pred}\right)^2}{n}} \tag{13}$$

$$R\% = \frac{n\left(\sum_{i=1}^{n} y_{i,\exp} \times y_{i,pred}\right) - \left(\sum_{i=1}^{n} y_{i,\exp}\right)\left(\sum_{i=1}^{n} y_{i,pred}\right)}{\sqrt{\left[n\left(\sum_{i=1}^{n} y_{i,\exp}\right)^2 - \left(\sum_{i=1}^{n} y_{i,\exp}\right)^2\right]\left[n\left(\sum_{i=1}^{n} y_{i,pred}\right)^2 - \left(\sum_{i=1}^{n} y_{i,pred}\right)^2\right]}} \times 100 \tag{14}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2}{\sum_{i=1}^{n}\left(y_{i,\exp} - y_{avg,\exp}\right)^2} \tag{15}$$

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2}}{y_{i,pred}} \tag{16}$$

where the prediction and experimental values are represented by $y_{i,pred}$ and $y_{i,\exp}$, whereas the sample's number is represented by n.

5.4 Performance Metrics

Throughout the experiments, the difference between the actual and the predicted stock closing price was used to calculate the RMSE, MSE, R and NRMSE values. The level of the stock-prices modification on the forecasting day was then calculated using the predicted data. A forecasting model, such as a time

series analysis or DL approach, is employed to forecast the future values of stock prices. This model takes historical data and potentially other relevant factors into account to generate predictions for future stock prices. The forecasting model produces a set of predicted stock prices for a specific time period, including the forecasting day of interest. These predictions are based on the model's understanding of patterns, trends and other factors observed in the historical stock-price data. To assess the level of modification or change in the stock price on the forecasting day, a comparison is made between the predicted stock price and the actual stock price observed on that day. This calculation aims to measure the difference between the expected value and the actual value. Stock prices may have significant variations in their scales, making it challenging to compare and analyze them directly. Data-normalization techniques, such as scaling or standardization, are applied to transform the data to a common scale, enabling fair comparisons and reducing the dominance of certain features in the prediction model.

Here, we used two different companies' datasets; namely, Apple and S&P 500 stock data. The collected data is processed by data normalization. First, we process S&P 500 stock data. Table 1 shows the comparison between Mobile U-Net V3 and Mobile U-Net V3 with BiLSTM, the hybrid techniques provide higher values for variables and give higher rank correlation values in the training phase. Table 2 illustrates the comparison of techniques for the forecast of closing prices in the testing phase. Figures 9 and 10 demonstrate the future prediction data values of the two datasets. Moreover, Figures 11, 12, 13 and 14 show the squared regression plots of the proposed method on the two datasets. Then, the proposed technique is compared with other recent research prediction techniques and the proposed method achieved higher prediction accuracy compared to other methods.

5.4.1 Results of S&P 500 Stock Data

The predictions of the S&P 500 closing price using the Mobile U-Net V3 and Mobile U-Net V3-BiLSTM models are shown in Table 4. The outcomes demonstrate the robustness and dependability of the DL method to forecast the closing price based on training. Based on the RMSE (0.01172) and NRMSE (0.02931) measures, Mobile U-Net V3-BiLSTM had a low prediction error rating.

Table 4. Results from deep learning for training-phase prediction of S&P 500 closing price.

Approach	MSE	RMSE	NRMSE	R%
Mobile U-Net V3-BiLSTM	0.00098	0.02978	0.07984	99.82
Mobile U-Net V3	0.00137	0.01172	0.02931	98.99

During the training stage, Figure 5 shows the exact combination between the predicted data of S&P 500's stock price and the actual values. The lack of a discernible difference between the actual and anticipated values serves as a proof of that both Mobile U-Net V3-BiLSTM and Mobile U-Net V3 were prepared to be evaluated during the training phase due to their high R percentages (99.82% and 98.99%, respectively) with extremely low RMSE and MSE values.

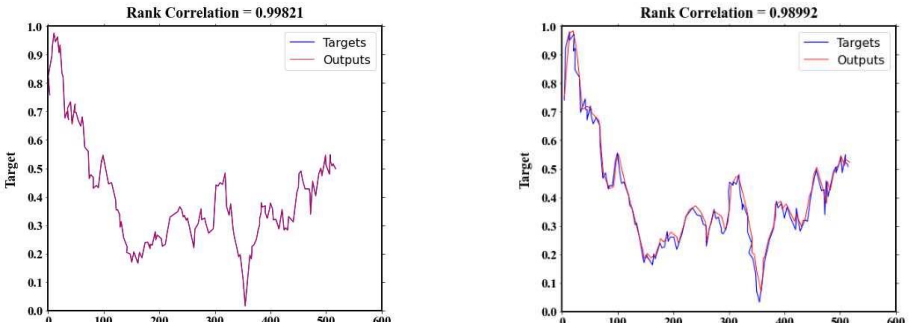

Figure 5. Ranking correlation between two DL techniques for forecasting the closing price of the S&P 500 stock market during the training stage: (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

The remaining 30% of the data were utilized, following training, to test the DL approaches' predictions shown in Table 5. The testing stage is crucial for assessing and determining the benefits of the proposed approaches for forecasting Tesla's closing price. The best outcomes were obtained using the Mobile U-Net V3-BiLSTM model (MSE = 0.0001057; RMSE = 0.01126).

Table 5. The outcomes for forecasting the closing price of the S&P 500 during the testing stage.

Approach	MSE	RMSE	NRMSE	R%
Mobile U-Net V3 - BiLSTM	0.00010	0.01126	0.03042	99.62
Mobile U-Net V3	0.00065	0.02624	0.07329	98.92

Figure 6 displays how the DL methods performed during the testing phase. Compared to Mobile U-Net V3 (98.92%), Mobile U-Net V3-BiLSTM has a higher R percentage (99.62%). The next step was to determine whether the prediction values of closing prices were in line with the actual values.

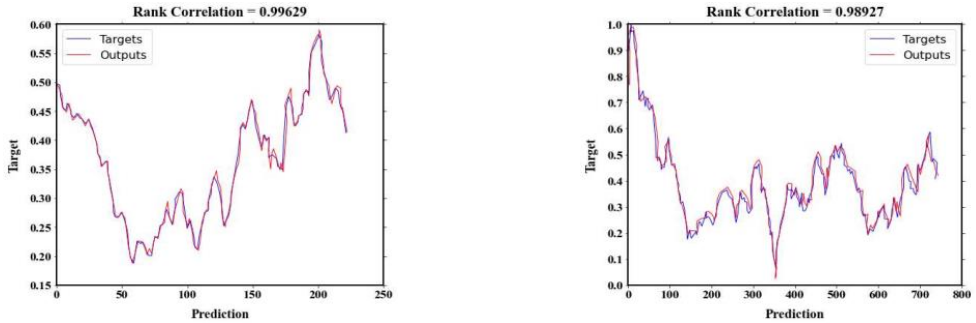

Figure 6. Ranking correlation between two DL approaches for forecasting the closing price of the S&P 500 stock market during the testing phase: (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

5.4.2 Results of Apple Data

After that, the DL algorithms were used to forecast Apple's stock-market closing price. Table 6 displays the outcomes of the DL approaches throughout the training phase (utilizing 70% of the data). According to the MSE measurements, the Mobile U-Net V3-BiLSTM model had a lower prediction of 3.894×10^{-5}.

Table 6. The outcomes for forecasting Apple's closing price due to training.

Approach	MSE	RMSE	NRMSE	R%
Mobile U-Net V3-BiLSTM	3.894×10^{-5}	0.00549	0.01843	99.93
Mobile U-Net V3	0.0001345	0.01075	0.0351	99.96

Figure 7 displays the effectiveness of the Mobile U-Net V3-BiLSTM and Mobile U-Net V3 models. The Mobile U-Net V3-BiLSTM model obtained a better R percentage (99.93%) according to the examination of the correlation metric. This demonstrates that the DL approach is suitable for forecasting future closing prices on the stock market.

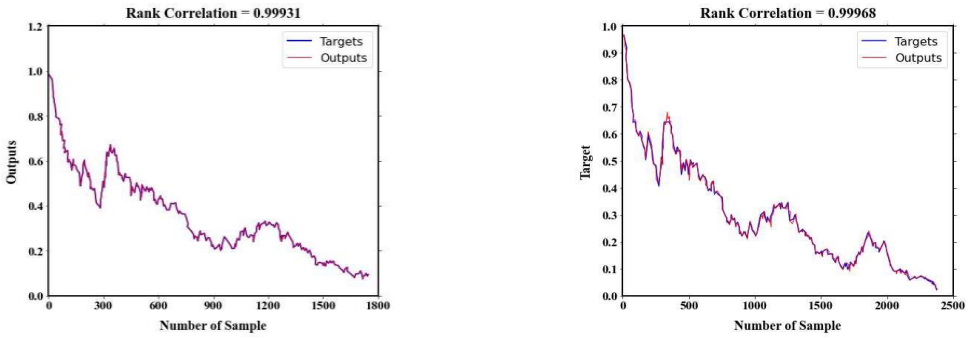

Figure 7. Ranking correlation between two DL techniques for forecasting Apple's closing price during the training stage: (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

The RMSE and MSE of the DL algorithms for forecasting the closing price of Apple due to the testing period are shown in Table 7 and Figure 8. The R percentage (99.87%) of Mobile U-Net V3-BiLSTM was relatively close to 100 even if the RMSE and MSE were the lowest feasible values. In this research, Mobile U-Net V3-BiLSTM had lower MSE and RMSE than Mobile U-Net V3. The R percentage for

Mobile U-Net V3 was 99.63% or nearly one. The results were somewhat better for Mobile U-Net V3-BiLSTM than for Mobile U-Net V3.

Table 7. The outcomes for forecasting Apple's closing price due to testing.

Approach	MSE	RMSE	NRMSE	R%
Mobile U-Net V3 -BiLSTM	3.894×10^{-5}	0.00549	0.01843	99.87
Mobile U-Net V3	0.0001345	0.01075	0.0351	99.63

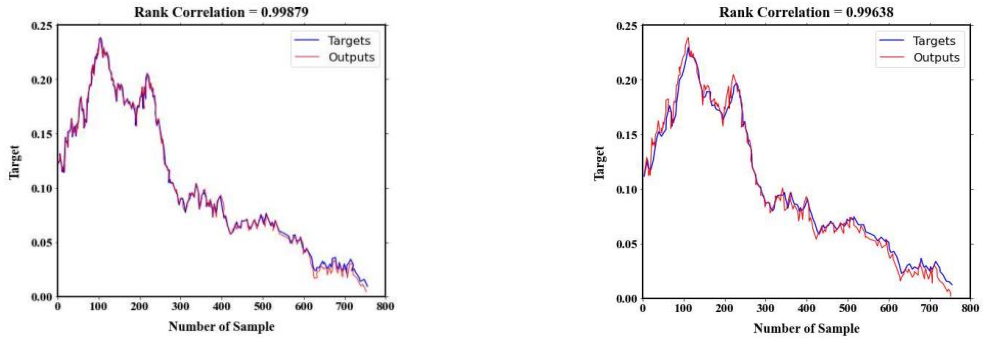

Figure 8. Ranking correlation between two DL approaches for forecasting Apple's closing price during the testing stage: (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

5.4.3 Predicting Future Values of S&P 500 and Apple Data

We predicted future values for S&P 500 and Apple from 28 February 2020 to 29 April 2020 to evaluate the performance of the Mobile U-Net V3-BiLSTM and Mobile U-Net V3 models (60-day period). Figure 9 demonstrates predicting the values of the S&P 500 Corporation utilizing the DL approach. Figure 10 shows the forecasting values for the Apple Corporation that were obtained using the deep-learning model.

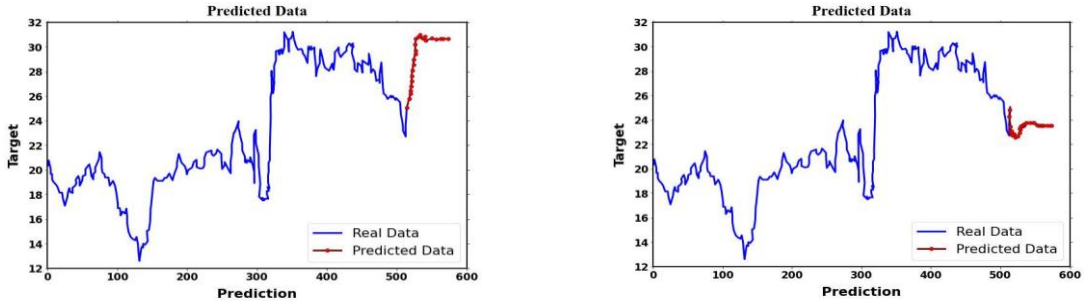

Figure 9. Predicting the future values of S&P 500 utilizing (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

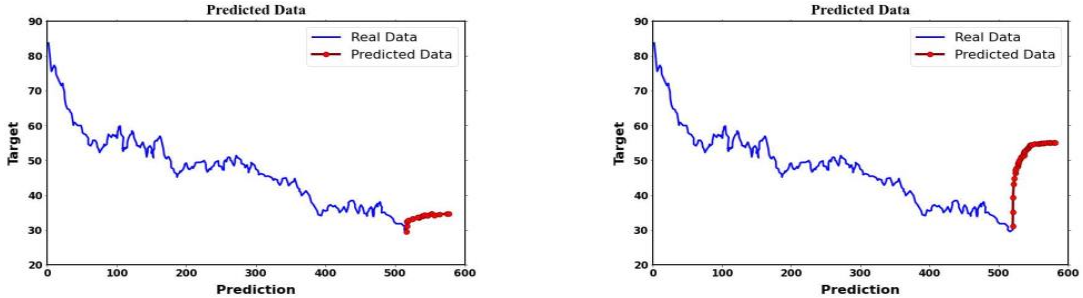

Figure 10. Predicting the future values of Apple utilizing (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3.

Investors have very simple access to more equities and stand to gain significantly from dividends paid as part of the corporation's shareholder-incentive strategy. On the stock market, investors desire to buy equities the values of which are anticipated to rise and sell those the values of which are anticipated to

decline. Stock traders must therefore be able to effectively predict the basic stock behavior before deciding to purchase or sell. The more accurate their prognosis of a stock's behavior, the more money they will profit from it. To help traders maximize their earnings, it is crucial to create an autonomous algorithm that can accurately predict market moves.

As a result, the capability of the DL techniques Mobile U-Net V3 and hybrid Mobile U-Net V3-BiLSTM to forecast S&P 500 and Apple stocks was examined in this paper. Figures 11 and 12 show, respectively, how these models performed throughout the training and testing phases for S&P 500. Both the training (Mobile U-Net V3-BiLSTM: R2 = 99.79%; Mobile U-Net V3: R2 = 98.76%) and testing (Mobile U-Net V3-BiLSTM: R2 = 99.31%; Mobile U-Net V3: R2 = 96.53%) stages of the evaluation saw higher Mobile U-Net V3-BiLSTM performance than Mobile U-Net V3. In comparison to Mobile U-Net V3, Mobile U-Net V3-BiLSTM thereby achieved more accuracy.

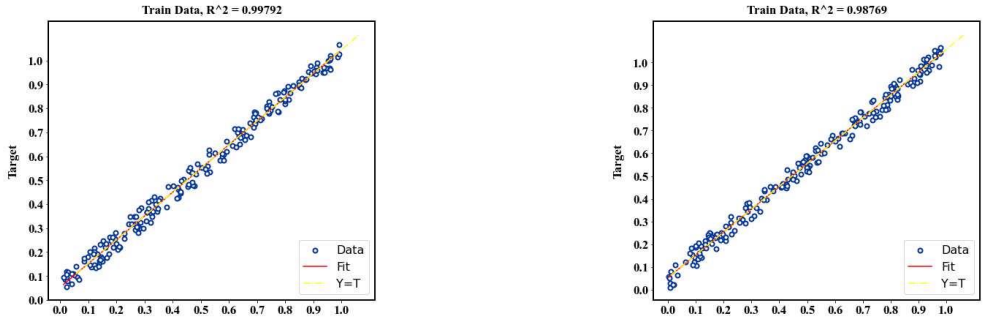

Figure 11. The squared regression plot of the proposed methods (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3 due to training for S&P 500.

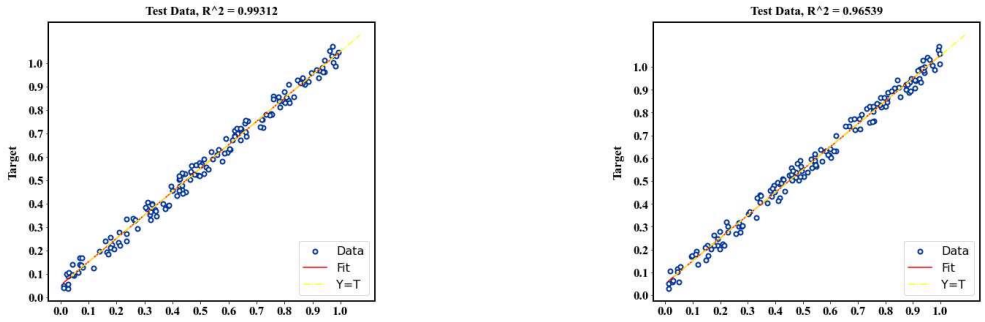

Figure 12. The squared regression plot of the proposed methods (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3 due to testing for S&P 500.

Figures 13 and 14 show that the predicted and actual values have a significant degree of agreement. Also, incredibly high R percent values were recorded in the training (Mobile U-Net V3-BiLSTM: 99.96%; Mobile U-Net V3: 99.82%) and testing (Mobile U-Net V3-BiLSTM: 99.72%; Mobile U-Net V3: 98.71%) phases. These values show that for the Apple data, the Mobile U-Net V3-BiLSTM model was more accurate and dependable than the Mobile U-Net V3 model.

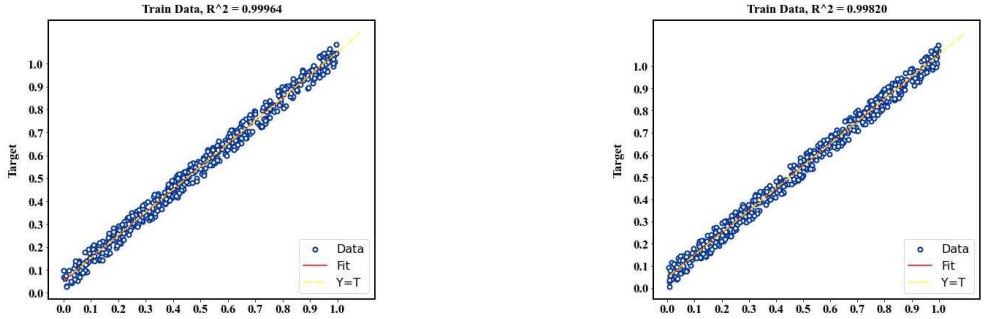

Figure13. The squared regression plot of the proposed methods (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3 due to training for Apple.

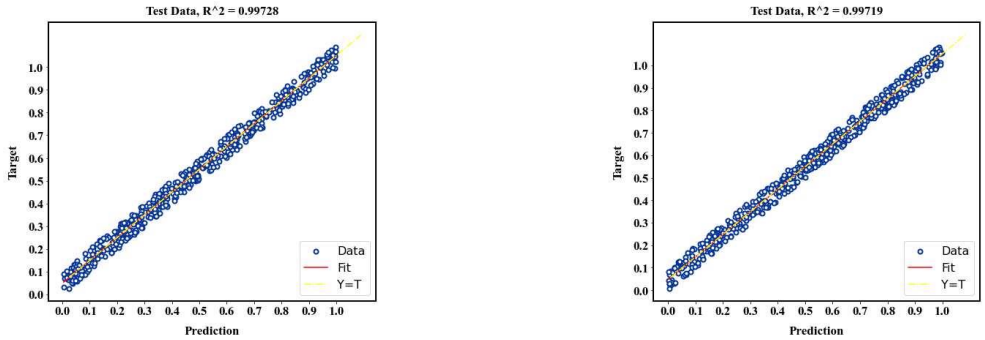

Figure 14. The squared regression plot of the proposed methods (a) Mobile U-Net V3-BiLSTM and (b) Mobile U-Net V3 due to testing for Apple.

We compared the outcomes of this proposed deep-learning framework with those of previous research to demonstrate Mobile U-Net V3's efficacy. Table 8 displays the outcomes of the Mobile U-Net V3 model provided in this work in comparison to the models used in earlier works. According to the MSE metric, the Mobile U-Net V3 model is superior to the models used in other research studies.

Table 8. Comparison of prediction outcomes of the proposed Mobile U-Net V3-BiLSTM system with the results of previous studies.

Source	Model	Dataset	MSE	R	RMSE
Aldhyani & Alzahrani [16]	CNN-LSTM	Tesla, Inc.	0.0001308	0.9926	0.01143
	CNN-LSTM	Apple, Inc.	5.725×10^{-5}	0.9973	0.00756
Yu & Yan [17]	LSTM	S&P 500	-	0.952	
Lu et al. [18]	CNN-BiLSTM-AM	Shanghai Composite Index stock	-	0.9804	31.694
Pang et al. [19]	ELSTM	Shanghai A-share composite index	0.017	-	-
Liu & Long [20]	LSTM	S&P 500	-	-	0.0075
Proposed Model	Mobile U-Net V3 – BiLSTM	S&P 500	0.000108	0.9962	0.9962
	Mobile U-Net V3 – BiLSTM	Apple, Inc.	3.894×10^{-5}	0.9987	0.9987

The proposed approaches may offer effective future prediction because of the possible benefits, which have long been a desire of most economies and people. Learning how to predict price changes in stocks might be helpful for those who are interested in studying stock-market forecasting. Predictions will be available to researchers that are more accurate than they have ever been because of artificial intelligence. Also, as technological advancements and algorithmic accuracy rise, its precision will rise with time. Our proposed method achieved higher accuracy value compared to other techniques. It is predict the closing price with less computation time complexity.

5.5 Evaluation of Training and Testing Set

As the number of iteration steps grows, graphs of loss value and prediction accuracy are shown in Figures 15 and 16. The graphs demonstrate the advantages results of implementing the study's proposed convergence technique. During the training phase, the proposed methods are trained for 200 iterations using the prepped training set. Currently, there is a 0.1 learning rate.

The training and testing accuracy, along with testing and training loss functions, are represented in Figures 15 and 16. 0.53 second is used for training the proposed model and 0.24 second is used for testing the proposed approach. During the training phase, the proposed method is trained for 200 epochs using the prepped training set. A learning rate of 0.1 has been established.

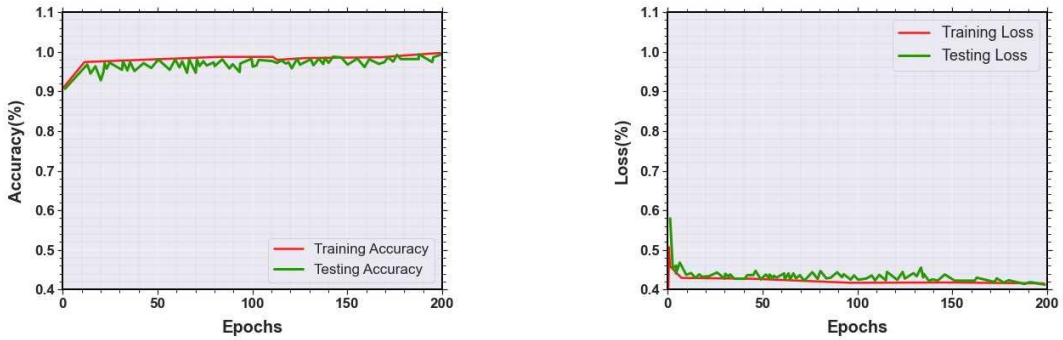

Figure 15. (a) Training and testing accuracy and (b) Training and testing loss for the Apple dataset.

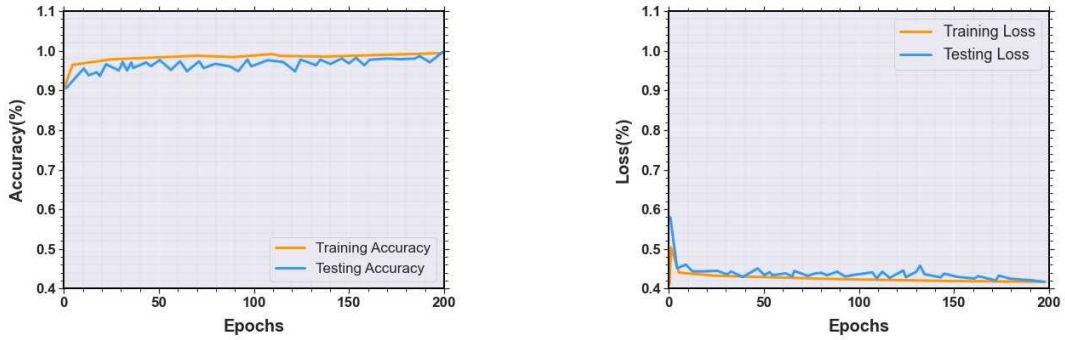

Figure 16. (a) Training and testing accuracy and (b) Training and testing loss for S&P 500 Stock dataset.

6. CONCLUSION AND FUTURE WORKS

The performance of companies, investor expectations, the geopolitical climate, investor perceptions and financial reports are just a few of factors affecting the stock market's behavior. When determining whether the stock price has increased or decreased, a company's profits are a crucial component to be considered. Also, predicting how the market will act for an investment can be challenging. To anticipate the stock-market analysis in this research, we used novel DL techniques. To normalize the input data given in the preprocessing stage, we are employing the min-max normalization approach. The stock-market analysis should then be predicted. Using of hybrid Mobile U-Net V3 and BiLSTM approaches, we predict the stock-market analysis using stock data from Apple, Inc. and S&P 500. The closing prices of these two firms are predicted. Our proposed method achieved a higher accuracy value compared to other techniques. Also, it is predicting the closing price with less computation time complexity. The evaluation of this research achieved a higher R-square value compared to other existing techniques. Further research will look at how well the model fits into various time-series prediction application fields, including forecasting gold prices, oil prices, earthquakes and weather, among others, based on AI techniques.

REFERENCES

[1] N. Jing, Z. Wu and H. Wang, "A Hybrid Model Integrating Deep Learning with Investor Sentiment Analysis for Stock Price Prediction," Expert Systems with Applications, vol. 178, no.1, pp.11-50, 2021.

[2] H. Rezaei, H. Faaljou and G. Mansourfar, "Stock Price Prediction Using Deep Learning and Frequency Decomposition," Expert Systems with Applications, vol. 169, no. 3, p. 114332, 2021.

[3] S. Mehtab, J. Sen and A. Dutta, "Stock Price Prediction Using Machine Learning and LSTM-based Deep Learning Models," Proc. of Symposium on Machine Learning and Metaheuristics Algorithms and Applications (SoMMA), pp. 88-106, Chennai, India, October 14–17, 2020.

[4] J. M. T. Wu, Z. Li, N. Herencsar, B. Vo and J. C. W. Lin, "A Graph-based CNN-LSTM Stock Price Prediction Algorithm with Leading Indicators," Multimedia Systems, vol. 23, no. 5, pp. 1-20, 2021.

[5] M. Vijh, D. Chandola, V. A. Tikkiwal and A. Kumar, "Stock Closing Price Prediction Using Machine Learning Techniques," Procedia Computer Science, vol. 167, no. 2, pp. 599-606, 2020.

[6] S. Mehtab and J. Sen, "A Time Series Analysis-based Stock Price Prediction Using Machine Learning and Deep Learning Models," Int. J. of Business Forecasting and Marketing Intelligence, vol. 6, no .4, pp. 272-335, 2020.

[7] Z. Jin, Y. Yang and Y. Liu, "Stock Closing Price Prediction Based on Sentiment Analysis and LSTM," Neural Computing and Applications, vol. 32, no. 5, pp. 9713-9729, 2020.

[8] S. Kumar Chandar, "Grey Wolf Optimization-Elman Neural Network Model for Stock Price Prediction," Soft Computing, vol. 25, no. 6, pp. 649-658, 2021.

[9] Y. Li, H. Bu, J. Li and J. Wu, "The Role of Text-extracted Investor Sentiment in Chinese Stock Price Prediction with the Enhancement of Deep Learning," Int. J. of Forecasting, vol. 36, no. 4, pp. 1541-1562, 2020.

[10] P. Gao, R. Zhang and X. Yang, "The Application of Stock Index Price Prediction with Neural Network," Mathematical and Computational Applications, vol. 25, no. 3, pp. 53-70, 2020.

[11] J. B. Awotunde, R. O. Ogundokun, R. G. Jimoh, S. Misra and T. O. Aro, "Machine Learning Algorithm for Cryptocurrencies Price Prediction," Proc. of Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities, pp. 421-447, 2021.

[12] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj and A. Iosifidis, "Using Deep Learning for Price Prediction by Exploiting Stationary Limit Order Book Features," Applied Soft Computing, vol. 93, no. 23, pp. 106-401, 2020.

[13] S. Mehtab and J. Sen, "A Robust Predictive Model for Stock Price Prediction Using Deep Learning and Natural Language Processing," arXiv preprint arXiv: 1912.07700, 2019.

[14] A. Ghosh, S. Bose, G. Maji, N. Debnath and S. Sen, "Stock Price Prediction Using LSTM on Indian Share Market," Proceedings of 32nd Int. Conf. on Computer Applications in Industry and Engineering, vol. 63, no. 33, pp. 101-110, 2019.

[15] X. Liang, Z. Ge, L. Sun, M. He and H. Chen, "LSTM with Wavelet Transform Based Data Preprocessing for Stock Price Prediction," Mathematical Problems in Engineering, 2019.

[16] T. H. Aldhyani and A. Alzahrani, "Framework for Predicting and Modeling Stock Market Prices Based on Deep Learning Algorithms," Electronics, vol. 11, no. 19, pp. 31-49, 2022.

[17] P. Yu and X. Yan, "Stock Price Prediction Based on Deep Neural Networks," Neural Computing and Applications, vol. 32, no. 3, pp. 1609-1628, 2020.

[18] W. Lu, J. Li, J. Wang and L. Qin, "A CNN-BiLSTM-AM Method for Stock Price Prediction," Neural Computing and Applications, vol. 33, no. 8, pp. 4741-4753, 2021.

[19] X. Pang, Y. Zhou, P. Wang, W. Lin and V. Chang, "An Innovative Neural Network Approach for Stock Market Prediction," The Journal of Supercomputing, vol.76, no.7, pp. 2098-2118, 2020.

[20] H. Liu and Z. Long, "An Improved Deep Learning Model for Predicting Stock Market Price Time Series," Digital Signal Processing, vol. 102, no. 11, pp. 102-741, 2020.

[21] X. Jin, J. Zhang, J. Kong, T. Su and Y. Bai, "A Reversible Automatic Selection Normalization (RASN) Deep Network for Predicting in the Smart Agriculture System," Agronomy, vol. 12, no. 3, pp. 55-91, 2022.

[22] P. Singla, M. Duhan and S.Saroha, "An Ensemble Method to Forecast 24-h Ahead Solar Irradiance Using Wavelet Decomposition and BiLSTM Deep Learning Network," Earth Science Informatics, vol. 15, no. 1, pp. 291-306, 2022.

[23] A. Alsenan, B. Ben Youssef and H. Alhichri, "MobileUNetV3—A Combined UNet and MobileNetV3 Architecture for Spinal Cord Gray Matter Segmentation," Electronics, vol. 11, no. 15, pp. 23-88, 2022.

[24] Apple, Inc. Stock Data, [Online], Available: https://www.kaggle.com/datasets/meetnagadia/apple-stock-price-from-19802021.

[25] S&P 500 Stock Data, [Online], Available: https://www.kaggle.com/datasets/camnugent/sandp500.

ملخص البحث:

نقترح في هذه الورقة نظاماً هجيناً يقوم على التّعلُّم العميق لتوقُّع الأسعار المستقبلية للأسهم. فقد حظيت تقنيات التّعلُّم العميق في الآونة الأخيرة بقدْرٍ كبيرٍ من الاهتمام؛ لأنّها أحرزت نجاحاً ملحوظاً في القدرة على التّعامل مع البيانات. ويستخدم النّظام الهجين المقترح بيانات السّوق من أجْل توقّع سعر الإغلاق. ولتجريب النّظام المقترح، تمّ تطبيقه على قاعدتي البيانات لشركة Apple, Inc. وشركة S&P 500. كذلك تمّت مقارنة النّظام المقترح بعددٍ من الأنظمة التي اقترحتها دراسات سابقة من خلال مجموعةٍ من المؤشّرات. وقد أثبت النّظام المقترح نجاعته وتفوُّقه على الأنظمة الأخرى من حيث الدّقّة في التّنبُّؤ بأسعار الأسهم.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

CONTAINER-BASED VIRTUALIZATION FOR BLOCKCHAIN TECHNOLOGY: A SURVEY

Nawar A. Sultan and Rawaa Putros Qasha

(Received: 29-Apr.-2023, Revised: 29-Jun.-2023, Accepted: 19-Jul.-2023)

ABSTRACT

Blockchain technology has garnered interest in several scientific and engineering fields. To improve blockchain-technology services, its execution challenges must be addressed. Container-based virtualization enables running isolated apps on a shared OS where blockchain technology can leverage this technology to run numerous nodes, smart contracts and decentralized apps in distinct containers allowing resource isolation and allocation, faster deployment and scalability and improved security through limited host OS and other container access. This article covers container-based virtualization for blockchain technology, including current methodologies, prospects and future perspectives. Initially, this study explains blockchain and containerization, as well as the reason for their integration. Then, reviews container virtualization services to address blockchain complexity, size, scalability and security. Conversely, container technology uses blockchain to protect data and enhance resource management. Next, it analyzes the latest containerization and blockchain integration studies. Finally, difficulties and future directions are considered to advance this promising research.

KEYWORDS

Container-based virtualization, Blockchain, Virtual machines, Docker, Kubernetes.

1. INTRODUCTION

In recent years, big data has gained immense significance, becoming a driving force behind the evolution of data processing, storage and management [1]. As a result, the demand for innovative technologies capable of meeting the growing challenges of big data has surged. This has led to the emergence and widespread adoption of containerization and blockchain, which have revolutionized the landscape of data management and security [2].

While blockchain technology offers decentralized and secure data management [3], container technology provides a simplified and scalable solution for system and application management [4]. The rise of cryptocurrencies like Bitcoin, Ripple and Ethereum has further propelled the need for blockchain technologies [1]. As such, the integration of containers with blockchain applications has become a critical area of research and development.

Blockchain technology can be implemented in virtual machines or containers, depending on the blockchain application's specific use case and requirements. Virtual machines can provide complete isolation between nodes in a blockchain network, making them a good choice for building private or permissioned blockchains, where security and data privacy are paramount. Each node can be deployed in a separate virtual machine with its operating system and resources, ensuring that any compromise or failure in one node does not affect the others [5].

Containers, conversely, are more lightweight and can be deployed more easily and quickly than virtual machines. They are a good choice for building decentralized applications that run on top of public blockchains, such as Ethereum. Each application can be packaged in a container and deployed to a decentralized network, where it can interact with other smart contracts and blockchain nodes [6]. In general, the choice between virtual machines and containers for blockchain applications depends on the application's specific use case and requirements. Security, scalability, portability and resource usage should be considered when choosing between these technologies [5].

Therefore, the most notable benefits of containerizing blockchain applications should be discussed to demonstrate the viability of doing so:

1) Efficiency in reducing costs through sharing resource features, where tens of containers can rapidly be virtualized to run on a single-core CPU [7].

N. A. Sultan and R. P. Qasha are with Department of Computer Science, University of Mosul, Mosul, Iraq. Emails: nawar.20csp62@student.uomosul.edu.iq and rawa_qasha@uomosul.edu.iq

2) Portability, containers are known to be genuinely built once and run software anywhere. This is achieved with the aid of platforms like Docker that facilitates the deployment process [8].
3) Improved security, as mentioned above, as containerization is an isolation technique; this allows apps to operate independently from each other and from the host system [9].
4) Agility and containerization facilitate the integration process, allowing developers to deliver the required enhancements rapidly [5].
5) Ease of management; with a container's orchestration platform like Kubernetes, it is easier to install, upgrade and manage containers [10].

Despite the growing interest and implementation of containerization in blockchain applications, there remains a need for a comprehensive study to address several shortcomings and challenges. This study aims to bridge the gap in the existing literature by exploring the benefits and limitations of containerizing blockchain applications, with a focus on improving the overall performance.

While blockchain technology and containers have many benefits, they also have restrictions and potential drawbacks when choosing these technologies for a particular use case. Careful planning and evaluation are necessary to ensure that the chosen technology meets the specific needs and requirements of the application. In the next sections, we focused on the importance of containers and blockchain technology and the role of containers in improving the performance of the blockchain. The literature review is structured as in Figure 1.

2. CONTAINER-BASED VIRTUALIZATION AND BLOCKCHAIN: AN OVERVIEW

In this section, we present a container-based virtualization overview, Docker and its orchestration mechanism, followed by an overview of blockchain technology.

2.1 Container-based Virtualization

Containers are lightweight executable packages for software codes that encapsulate these software codes with only the required operating-system libraries and dependencies in a way that abolishes the need for a specific infrastructure to run [11]. Over the last few years, containers gained maturity and popularity because of the lightweight virtualization that they provide, which enabled multiple applications to run independently and isolated from any other application and from the operating system and without the need for occupying the operating-system kernel entirely for each one of them, as it is the case with virtual machines [12]. Rather, containerization enables sharing the host operating-system kernel with multiple containers simultaneously. This sharing feature utilizes resource employment in a way that reduces the amount of the required hardware resources. Moreover, containers are easier to manage than VMs (virtual machines), thanks to their orchestration engines like Kubernetes, which influenced the emergence of many container cloud platforms [13]. Containerization, which is container virtualization, refers to creating an isolated virtual environment for each application, which is directly related to kernel functionalities. Figure 2 shows the process of shifting from VMs to containers, where each virtual environment is named a container and both namespaces and (cgroups) refer to functions provided by the operating-system kernel. The namespaces control and limit each process's number of resources used, while (cgroups) deal with a process group's resources [14].

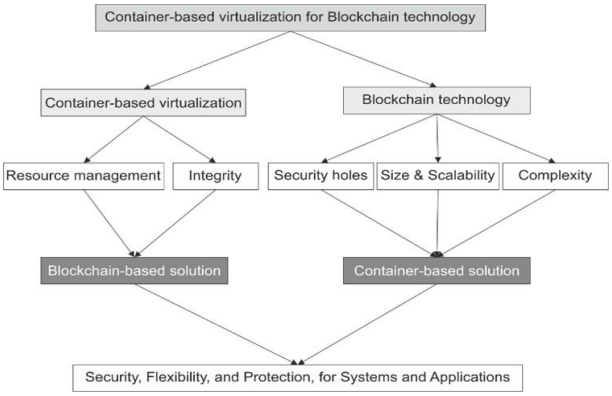

Figure 1. The structure of the literature review.

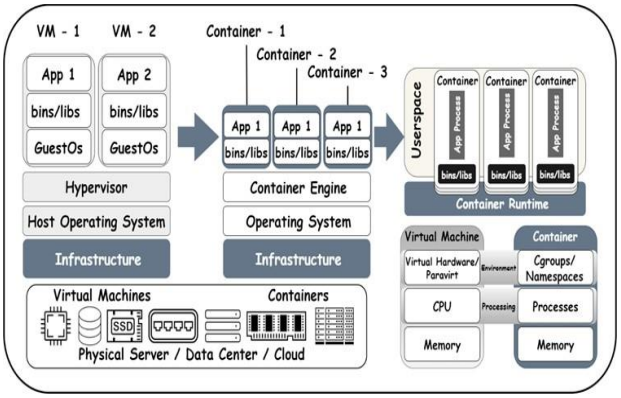

Figure 2. From virtual machines to containers.

System solutions based on hypervisor suffered many drawbacks related to resource allocation, start-up time and adaptability [15]. Currently, Docker is the most widely used virtualization tool, because it significantly increases operating system resource usage with little additional overhead. The open-source containerization engine called Docker automates the packing, shipping and deployment of software applications. These programs are delivered as thin, portable, self-contained containers that may operate almost anywhere. A Docker container is a program container that includes the constituents required for the program to run automatically [16]. A single system may contain several Docker containers and each container is totally independent of the host machine. Particularly, the software component and all of its needs are included in a Docker container, such as binaries, libraries, configuration files, scripts, jars, …and so on [17]. The following elements make up the majority of the Docker solution:

1) The Docker engine.
2) The Docker hub.

The Docker engine makes it possible to realize both general-purpose and purpose-specific Docker containers. A fast way to expand the set of Docker phases that may be gathered in different ways to make publicly-discoverable, network-accessible and exceedingly-consumable containers is *via* the Docker hub [14].

Let's say that we wish to execute the containers directly on a Linux computer. The diagram in Figure 3 shows how the Docker engine creates, oversees and manages many containers [17].

Container-based Docker ecosystem has the following characteristics [18]:

1) It supports portability by allowing for the packaged app to run anywhere, since it facilitates and improves the processes of the application's development and deployment, which makes it easy to build, ship and run any app and everywhere.
2) It strengthens the integrity of any infrastructure by enabling developers to package the application with all its necessary libraries and dependencies to build up workable software that works properly in any environment without the need for any prior setup.
3) It is easy to manage Docker by anyone in a way that meets the required additional features [19].

The Docker engine, which creates and runs containers and the Docker hub, which presents a cloud service for distributing containers, are the two distinct parts of the Docker platform [20].

Few Docker containers on a single system are easy to manage, but challenges arise when having to put these containers into production on a dispersed host network; therefore, tools for ensuring availability, scaling, networking, integration and administration are essential for managing dynamic containers as one entity on a network, where manual handling is impossible in this case [13].

Therefore, tools like Google Kubernetes, Marathon (a framework for Mesos), CoreOS's Fleet and Docker's swarm tooling are essential for managing containers on a network system. Each container needs on-host placement, monitoring and updating and at the same time, the system must be enabled to respond to failures, loading or any change in the system by taking the appropriate action by either moving, starting or aborting any container [21].

2.2 Blockchain Technology

Blockchain is a decentralized and distributed digital ledger that records transactions and stores data across a network of computers. It operates using a consensus mechanism that allows multiple parties to verify and agree on the validity of transactions without needing a trusted intermediary or central authority [22].

Each block contains a record of several transactions, as well as a unique cryptographic hash that identifies the block and links it to the previous block in the chain [23]. This creates an immutable and tamper-proof record of all the transactions on the blockchain [24].

One of the key benefits is that blockchain enables trust and transparency in a digital world without the need for intermediaries. Transactions are verified and recorded by a network of nodes, each with a copy of the blockchain. This ensures that any attempted tampering or fraud is easily detected and prevented [25].

Another key benefit of blockchain technology is that it can be used to create smart contracts; self-executing contracts with the terms of the agreement written into the code (see Figure 4). Smart contracts can automate the execution of transactions and eliminate the need for intermediaries, reducing costs and increasing efficiency [26].

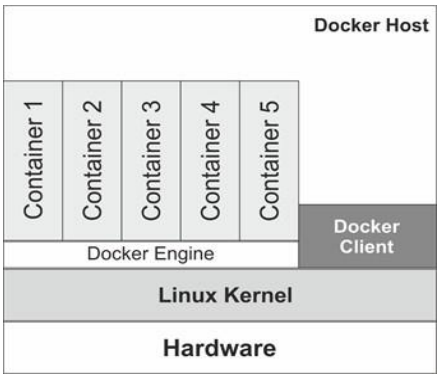

Figure 3. Docker engine.

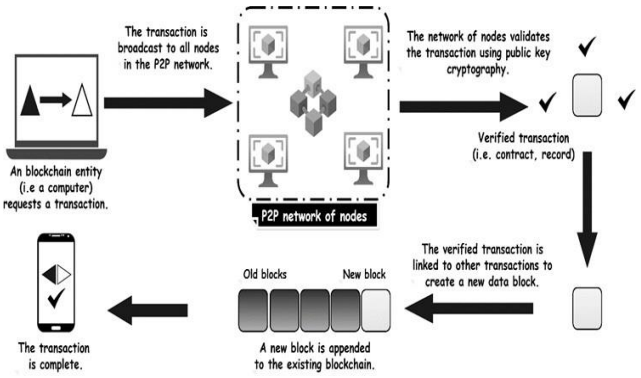

Figure 4. Contract creation by blockchain.

Blockchain technology has many potential use cases, including:

Cryptocurrencies: The underlying technology behind cryptocurrencies includes Bitcoin and Ethereum. These digital currencies use blockchain to enable secure, decentralized transactions without the need for intermediaries, like banks or financial institutions [27]. Supply-chain management: can track and verify the movement of goods and products across a supply chain. This can increase efficiency, reduce fraud and improve transparency in the supply chain [28].

Voting systems: Blockchain technology can create secure and transparent voting systems that eliminate the risk of tampering or fraud [29]. Identity verification: Blockchain technology can create decentralized and secure identity verification systems that eliminate the need for intermediaries, like government agencies or financial institutions [30]. However, blockchain technology also has some limitations and potential drawbacks. These include [31]:

Scalability: Blockchain technology can be slow and resource-intensive, especially when running on a large scale.

Energy consumption: The process of verifying transactions on a blockchain can be energy-intensive, leading to concerns about the environmental impact of blockchain technology.

Specialized infrastructure and software: Blockchain technology requires specialized infrastructure and software to run effectively, which can increase the cost and complexity of deployment.

Overall, blockchain technology represents a promising and innovative approach to sharing and verifying information in a secure and decentralized manner. While it has some limitations and potential drawbacks, careful evaluation and planning can help ensure that blockchain technology is used effectively and appropriately for a given use case [32].

3. Motivation for Integrating Container-based Virtualization and Blockchain

The challenges of adopting container-based virtualization, the technical constraints of blockchain and the promising opportunities of combining such two technologies are highlighted in this section as the driving forces behind the integration of container-based virtualization and blockchain technologies.

3.1 Challenges with Container-based Virtualization

Despite being the next revolution in cloud computing, containers are lighter than virtual machines. In comparison to conventional VMs, they may significantly reduce the start-up time for instances, as well as the processing and storage overhead [12]. However, like many other things, using and deploying containers presents particular difficulties for developers. CNCF Survey 2020, published in https://www.cncf.io/reports/cloud-native-survey-2020/, revealed that development teams encountered difficulties with several containerization-related issues (Figure 5).

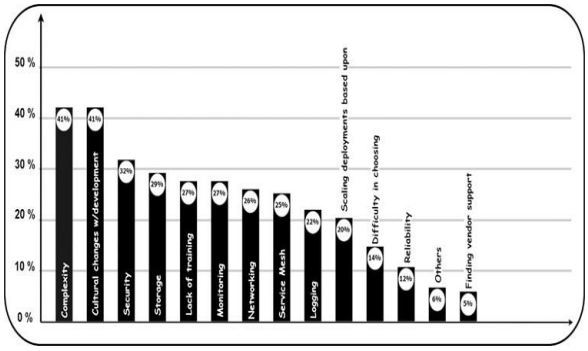

Figure 5. Challenges in using/deploying containers.

The survey showed that for (41%) of respondents, complexity was recorded to be the top challenge of containers' usage and deployment. Security came in the second place and recorded (32%) of respondents, followed by storage (29%) and lack of training and monitoring (both at 27%).

3.2 Technical Limitations of Blockchain

Despite its numerous advantages, blockchain technology has some technical limitations that need to be addressed [31]:

Scalability: As the number of transactions increases, the size of the blockchain grows, making it difficult to store and process large amounts of data. This issue becomes particularly problematic for public blockchains that require nodes to store the entire history of the chain [2].

Speed: The time it takes to validate transactions and add them to the blockchain can be slow, particularly for large networks. This is due to the fact that each transaction needs to be verified by multiple nodes in the network [33].

Cost: The cost of running a node on the network and participating in the consensus process can be high.

This is particularly true for proof-of-work blockchains, which require significant amounts of computing power and energy [25].

Interoperability: Different blockchain networks use different protocols, which can make it difficult for them to communicate with one another. This limits the potential for blockchain to be used for large-scale applications that require interoperability between different systems [34].

Security: While blockchain is considered to be a secure technology, it is not immune to hacking or other security breaches. Additionally, the use of smart contracts on blockchain networks can introduce additional security risks if they are not properly coded or audited [35].

Governance: The decentralized nature of blockchain can make it difficult to reach a consensus on changes to the network. This can lead to governance challenges and delays in implementing necessary updates or improvements [36].

3.3 The Integration of Container-based Virtualization and Blockchain Possibilities

There are several potential possibilities for integrating blockchain and container-based virtualization, some of which are outlined below:

Immutable container images: Blockchain technology can create a tamper-proof and immutable ledger of container images. This can help ensure the authenticity and integrity of container images and prevent unauthorized modifications or tampering [20].

Smart contract-based container management: Smart contracts can be used to automate the deployment and management of containerized applications. This can help simplify the container-management process and reduce the risk of errors and misconfiguration [37].

Also, to improve the scalability of blockchain technology, here are some possibilities:

Containerized blockchain nodes: Using container-based virtualization, deploying multiple blockchain nodes on a single machine or cluster of machines is possible. Distributing the workload across multiple nodes can help improve the blockchain network's performance and scalability [33].

Consensus algorithm optimization: Container-based virtualization can be used to optimize the consensus algorithm of the blockchain network by deploying multiple nodes with different consensus algorithms and testing their performance. This can help identify the optimal consensus algorithm for the blockchain network [38].

Overall, integrating container-based virtualization and blockchain technology can help improve blockchain networks' scalability and performance, enabling them to handle increased traffic and usage and support large-scale applications.

3.4 Feasibility of Integrating Container-based Virtualization and Blockchain

The integration of container-based virtualization and blockchain is a feasible and promising approach to improve the scalability, security and flexibility of blockchain networks. Container-based virtualization can help isolate different components of the blockchain network, allowing them to run in separate containers and reducing the risk of vulnerabilities or attacks affecting the entire network [7].

Furthermore, by using container-based virtualization, it is possible to deploy multiple instances of blockchain nodes with different consensus algorithms, allowing for easier testing and optimization of the blockchain network's performance. This approach can also enable more efficient resource utilization and better management of the network's computational resources [38].

While challenges must be addressed, the feasibility of integrating container-based virtualization and blockchain is high. The potential benefits make it a promising approach for the future of blockchain technology.

4. LITERATURE REVIEW

Through this survey, we touched on the strengths and weaknesses of both technologies and what the best solutions and procedures were that could be followed to achieve the best integration between them. There is a lot of research on the blockchain or its services, specifically given that the technology is open-source and has been used in bitcoins for about fourteen years. Still, there are only a few papers that discuss how to improve blockchain using containerization technology or how they can be incorporated. We have noticed through previous studies that there are major weaknesses in blockchain technology that can be addressed or improved through container technology, which can be summed up into three groups (complexity, size and scalability, and security holes).

4.1 Complexity

Zeadally and Abdo [39] [24] provided a performance-evaluation mechanism to aid in the decision-making of blockchain-based service planners. For portability and flexibility, this system is available as Docker and Kubernetes. This study's experimentation method enables service providers to assess the server performance necessary for the service's launch. Additionally, because of Docker and Kubernetes power, the experimental environment enables creators to create scenarios, like deploying several servers and replicating as many client pods as they like.

This study presents the initial review by Chen, C. et al. [6] on how decisions in blockchain design affect performance. Then, they provided their approach for employing containerization to test blockchains. In this approach, authenticity and the expensive test of P2P applications were balanced. Finally, they put their framework into action to perform a demo assessing how the characteristics of Bitcoin's network would affect system dependability. Because of this, they discovered three benefits of employing containerization for blockchain: authenticity, ease of deployment and low cost.

Minichain, a container-based emulator for testing blockchains utilizing a proof-of-work mechanism, was proposed by Wu, X. et al. [39]. Minichain offers a realistic and adaptable network environment which is absent from current blockchain experiments.

A realistic study is expensive and time-consuming, because no agreed paradigm exists for evaluating permissioned blockchain platforms' scalability and comparing consensus algorithms' performance and features. Mazzoni, M., Corradi A. and V. Di Nicola examined the Quorum blockchain's functional scalability and application [35]. They presented a framework for any permissioned blockchain technology, even though they reviewed a financial use case. They chose Hyperledger Caliper for

benchmarking and Docker for deployment for repeatable, cross-platform and cost-effective analysis.

4.2 Size and Scalability

Pongnumkulet et al. [40] conducted a study, in which two famous blockchain platforms were examined in terms of performance: Hyperledger Fabric and Ethereum. Hyperledger Fabric a private (permissioned) blockchain implementation, is designed to serve as a building block for blockchain applications in a variety of sectors. As a result, its design is modular, enabling plug-and-play compatibility for parts, like consensus and membership services. To assess the performance and constraints of these cutting-edge platforms, the study used container technology (Docker) to allow Ethereum (private deployment) and smart contracts, also known as "chaincode," which make up the application logic of the system, because one common objection to existing blockchain technology is its inability to scale. Consequently, there are two objectives for this preliminary performance review. The development of a blockchain-platform evaluation methodology comes first. The analytical results are also shared with practitioners to help them understand how to integrate blockchain technology into their IT systems. According to the testing results, which were based on different quantities of transactions, Hyperledger Fabric regularly surpasses Ethereum in terms of execution time, latency and throughput.

Current blockchain designs need to scale due to exponential message and memory complexity. Researchers presented (Hassanzadeh et al. [41]) LightChain, the first fully decentralized Distributed Hash Table (DHT)-based blockchain architecture, to address operational scalability challenges. They created a containerized model of a LightChain node system that can be run on a single computer, improving repeatability.

To address blockchain workload changes. Z. Shi et al. [42] recommended the high-performance Kubernetes scheduling technique HPKS for offline workload management. HPKS reduces worker node consumption by 13.0% in PoS blockchain applications. Compared to Kubernetes' default scheduler, Makespan's HPKS increase is less than 3%.

Volpe et al. proposed a blockchain-based smart-contract architecture for manufacturing digital processes [43]. Their key contribution is integrating blockchain with Cloud Storage and Docker, two well-known technologies.

4.3 Security Holes

Using W3C-PROV Data Model, El Ioini, N. and Pahl, C. [44] suggested a container-based blockchain architecture that tracks the sources of all orchestration choices made by a business network. This architecture offers fresh ways for many parties to communicate, enabling secure transactions and creating a new decentralized interaction paradigm for IoT-based applications.

A thorough analysis of blockchain-based trust techniques in cloud-computing systems was carried out by Li, W. et al. [18]. Using a novel method of cloud edge trust management and a cloud transaction model based on double-blockchain structures, they were able to identify the remaining challenges. They offered suggestions for additional research in this area.

Concerning container technology, we noted its reliance on blockchain technology to protect container data from tampering and maintain its integrity, in addition to making use of it to improve the efficient resource-management process.

Brinckman et al. [16] conducted a study, where many applications in research, science and industry have been stimulated by the introduction of such lightweight environments (containers), making it possible to share, reuse and instantiate pre-configured operating environments as needed. Currently, centralized repositories (such as Docker hub) have made it possible to share containers, which serves as the foundation for future growth. The researchers look at whether a distributed group of users can safely exchange container- based programs and provide an audit trail showing what has been shared and with whom. They conducted a comparison of blockchain technologies for this use case while taking into account the features of these blockchain technologies for this purpose. The majority of research was reviewing and categorizing various ledger systems.

Tosh et al. [45] stated that cloud data provenance must be secure against malicious actors. The researchers proposed a blockchain-based data provenance architecture (BlockCloud) that integrates a

proof-of-stake (PoS)-based consensus protocol to securely record data operations in a cloud environment powered by a novel PoS consensus model (CloudPoS). Validators are cloud-computing stakeholders. Rewarding involvement or securing collaboration may drive such engagement. Because participants' resources are at stake, the suggested leader-election procedure gives every cloud user an equal chance of leading the Blockchain based on the number of resources staked. Hence, blockchain activity is safer.

In a study by Marko et al. [37], resource-use optimization in a commercial setting is studied using a home- built video-conferencing (V.C.) system. All parties involved, including end users, cloud-service providers, software developers, …etc., are not provided with monetization alternatives by this type of application. Related to the technology of blockchain, Smart Contracts (SCs) may be able to help with some of these requirements. A unique architecture is provided for monetizing value generated in accordance with the desires of the stakeholders who take part in joint software service offers.

A revolutionary "permissioned Hyperledger Fabric blockchain containerized cloud ecosystem" was proposed by Awuson-David et al. [46] to protect and maintain the veracity of digital proof during both storage and transmission. Then, a "Dockerized private blockchain cloud ecosystem architecture" was designed and implemented, which would decrease the challenges faced by forensic investigators in the cloud ecosystem by ensuring evidence integrity in a multi-tenancy, private cloud environment.

J. Sun et al. [20] depending on blockchain technology, developed a container cloud security system in response to the susceptibilities and malware in container imageries in addition to specific incorrect settings that breach security-compliance standards.

Marques et al. [48] [14] described how containerization's flexibility made system monitoring harder due to the huge volume of calls and (de) allocations. This study investigated how documenting these activities in a blockchain-based data structure could simplify resource audits and procedure-order analytics. Blockchains allow container-based solution creators, end users and vendors more trust in record integrity. To meet their needs for security, flexibility and protection, numerous systems and applications have benefited from the integration of the two technologies.

Vorakulpipat, C. and Chaisawat, S. [29], in order to provide data security and flexibility in system integration, developed a system design architecture. By monitoring the use of computer resources and performing performance tests on the design, it was further examined and evaluated to confirm that it complied with the aforementioned requirements.

Aujla, G. S. et al. [47] developed a blockchain-based secure data processing system that creates a multi-objective optimization problem for an edge-envisioned vehicle-to-everything (V2X) scenario. It also features an ideal container-based data-processing scheme and a blockchain-based data integrity-management scheme to reduce latency and connection breakage.

According to Kumar, P. and Shah, M. [48], obtaining an accurate birth certificate for any individual is a significant challenge. The researchers presented and created a birth certificate storage system based on the "InterPlanetary File System" (IPFS) and "BLOCKCHAIN" technology in this paper. Due to the advancement of Docker technology and containerization as a service, this application was also deployed inside a container using Docker-compose, which creates a multi-container Docker application (CaaS).

Table 1 summarizes the issues that researchers face when fusing blockchain with containers, along with solutions for each technology.

Table 1. Literature review.

Ref.	**Research Problem**	**Challenges Faced by Blockchain/Containerization**	**Solutions Provided by Blockchain/ Containerization**
[39][24]	Portability and flexibility in the blockchain	Measuring the performance that aids in decision-making for service planners using blockchain technology	Docker and Kubernetes for portability and flexibility
[16]	Sharing container-based applications securely	Sharing container-based programs safely across a decentralized group of users while keeping track of who has shared what and with whom by using an audit trail	Blockchain technologies

[40]	Scalability of blockchain technology.	Methodology for evaluating a blockchain platform	Containerization-based
[6]	Testing blockchains	Traditional testing has issues with being unconvincing or expensive	Using containerization for Blockchain for testing, easing deployment, lowering cost and authenticity
[45]	Preserving tamper-resistant data provenance in the cloud	Making containerisation more secure	Blockchain-based data provenance architecture
[44]	Trustworthy transactions	Identifying each device/data in the network and tracking the provenance of its actions	Blockchain container-based architecture
[39]	Testing proof-of-work-based blockchains	A platform for realistic testing and evaluation of blockchain systems and applications	Container-based emulator
[37]	Providing monetization possibilities to all involved stakeholders in the video-conferencing (VC) system	Optimization of resource use for container-based video conferencing in a business context	Smart contracts
[46]	Integrity and confidentiality of data in a cloud environment	Reducing the difficulty of acquiring evidence in the cloud	Dockerized private blockchain cloud ecosystem architecture
[20]	Viruses and vulnerabilities in container images	Container cloud security enhancement	Systembased on blockchain technology
[41]	Challenges with scalability in blockchain architectures	Due to its asymptotic message and memory complexity, blockchain architectures face scalability issues	A containerized LightChain system proof-of-concept implementation
[29]	Delivering data security and agility in system integration		Using container technology along with blockchain technology
[47]	Securing data processing	The privacy of user data/activities	Using container technology along with blockchain technology
[48]	Identifying the correct birth certificate of any person		Using container technology along with blockchain technology
[18]	Building a trust-enabled transaction environment		A double- blockchain structure-based cloud transaction model
[43]	Integration of blockchain with containerization and cloud storage	Collaboration in the cloud when offering and consuming different services is a shortcoming of the blockchain	Docker and cloud storage
[48] [14]	Flow of calls and (de)allocations in massive amounts in container-based virtualization	System monitoring in container-based virtualization	Blockchain-based solution
[42]	Addressing the characteristics of PoS(Proof of Stake) blockchain applications in the cloud	Workload changes in blockchain applications	Kubernetes container orchestration
[44]	Choosing which blockchain technology and consensus algorithm best fit a specific use case is complex	Permissioned blockchain platforms need a common framework for evaluating scalability	Docker as a deployment tool

4.4 Shortcomings of Previous Studies

Previous studies' efforts in the field have provided valuable insights into the application of blockchain and container technologies. However, these studies have often lacked a detailed examination of the specific challenges associated with integrating containers into blockchain environments. Consequently, there is a need for a more focused investigation to address the following shortcomings:

1) Limited exploration of performance enhancements: Previous studies have primarily focused on the general benefits of containerization and blockchain technology without thoroughly investigating how containerization can improve the performance of

blockchain applications. The impact of containerization on scalability, resource utilization and overall efficiency requires deeper analysis.

2) Inadequate examination of security considerations: While containerization offers enhanced security through isolation, there is a need to explore the potential vulnerabilities and risks associated with deploying containerized blockchain applications. A comprehensive understanding of the security implications is crucial for ensuring the integrity and privacy of blockchain networks.
3) Insufficient evaluation of deployment and management challenges: Previous studies have often overlooked the complexities involved in deploying and managing containerized blockchain applications. The effective orchestration of containers, integration with blockchain networks and efficient resource allocation require careful consideration to maximize the benefits of containerization.

By addressing these limitations and delving into the specific challenges associated with containerizing blockchain applications, this study aims to provide a comprehensive understanding of the potential improvements that can be achieved. The following section will explore the benefits of containerization, such as cost reduction, portability, enhanced security, agility and ease of management, thereby highlighting the viability and significance of integrating containers with blockchain technology. Table 2 summarizes the aims and rationale of this survey in the context of previous works.

Table 2. Aims and rationale of the current survey.

Research Aspect	**Previous Works**	**Aim and Rationale**
Aim and Rationale	Limited exploration of containerization's impact on blockchain performance	Investigating how containerization can enhance scalability, resource utilization and overall efficiency in blockchain applications
Security Considerations	Inadequate examination of security implications of containerized blockchain applications	Analyzing the potential vulnerabilities and risks associated with deploying containerized blockchain applications to ensure data integrity and privacy
Deployment and Management	Insufficient evaluation of deployment and management challenges	Exploring the complexities involved in deploying and managing containerized blockchain applications, focusing on effective orchestration, integration with blockchain networks and efficient resource allocation

5. CONTAINER-BASED VIRTUALIZATION ROLES FOR BLOCKCHAIN ENHANCEMENT

Container-based virtualization enhances blockchain technology by addressing various research issues and providing valuable solutions. The following points provide a more detailed elaboration on the research issues related to container virtualization for blockchain technology:

Scalability: One of the key challenges in blockchain networks is scalability. As blockchain networks grow, the number of transactions being processed increases and the scalability challenge arises in ensuring that the network can handle the expanding workload efficiently. Public blockchains, in particular, face scalability concerns due to the requirement of storing the entire history of the chain on every participating node [31]. To address scalability challenges, various techniques have been explored, such as optimizing consensus algorithms, implementing sharding mechanisms or introducing layer-two scaling solutions, like state channels or sidechains [35]. These approaches aim to improve the throughput and capacity of blockchain networks, enabling them to handle a larger number of transactions or computations per unit of time. Deploying blockchain nodes within containers enables more efficient scaling of the network. By utilizing containerization, blockchain networks can dynamically adjust their capacity to meet fluctuating demands. Containers offer lightweight, portable environments that can be easily replicated and moved between hosts. This enhances the scalability of blockchain networks, allowing them to expand or contract as needed [2].

Security: It is a paramount concern in blockchain networks. Containerization contributes to enhancing the security of blockchain networks in multiple ways. By isolating blockchain nodes within separate containers, the impact of potential attacks or compromises is limited. Each container acts as an

independent unit, reducing the likelihood of an attacker gaining access to the entire network. Additionally, containerization simplifies the deployment of security patches and updates across the network, ensuring that the latest security measures are promptly applied [35].

Consistency: It is crucial for the reliable functioning of blockchain networks. Containerization aids in achieving consistency by utilizing container images. These images ensure that all nodes within the blockchain network are created with the same software stack and configurations. This reduces the risk of configuration errors and inconsistencies that can compromise the integrity of the network. Managing and maintaining the blockchain network become easier as containers offer a standardized and reproducible environment [42].

Deployment: Efficient deployment of blockchain networks is another critical research issue. Containerization, coupled with container orchestration tools like Kubernetes, simplifies and automates the deployment process. These tools enable the seamless distribution of blockchain nodes across multiple hosts, saving time and reducing the risk of human errors during deployment. Containerization streamlines the setup and configuration of blockchain networks, enhancing their manageability and overall deployment efficiency [49].

Overall, container-based virtualization provides solutions to critical research issues related to scalability, security, consistency and deployment in the context of blockchain technology. By leveraging containerization, blockchain networks can achieve enhanced performance, security and manageability. Using containerization technologies and methodologies enables more reliable and efficient utilization of blockchain networks, paving the way for their widespread adoption in various industries.

6. ISSUES AND CHALLENGES

Integrating container-based virtualization and blockchain technology can present several issues and challenges that must be addressed.

Security: While containerization can enhance the security of blockchain networks, it can also introduce new security risks. Containers can be compromised if they are not properly secured or if vulnerabilities in the container image are exploited. This could compromise the security of the entire blockchain network [9].

Performance: Containerization can affect the performance of blockchain networks. Running blockchain nodes in containers can result in overhead, which can impact the speed and throughput of the network. Careful optimization is required to minimize this overhead and ensure optimal performance [11].

Complexity: Integrating container-based virtualization and blockchain technology can add complexity to the deployment and management of blockchain networks. Container orchestration tools like Kubernetes can help simplify this process and introduce new layers of complexity that need to be managed [35].

Compatibility: Compatibility between container-based virtualization and blockchain technology can be an issue. Not all blockchain platforms may be compatible with containerization or require specific configurations to work effectively in a containerized environment [5].

Overall, integrating container-based virtualization and blockchain technology requires careful consideration and planning to address these challenges and ensure that the resulting network is secure, performant and manageable.

7. CONCLUSIONS

Previous studies have focused on the general benefits of containerization and blockchain technology, but have not extensively explored the challenges associated with integrating containers into blockchain environments. While some research has touched upon the advantages of containerization, such as resource sharing, portability, security, agility and ease of management, there is a lack of detailed analysis regarding the limitations and potential risks involved. Additionally, the performance enhancements achieved through containerization in the context of blockchain applications have not been adequately examined. Furthermore, studies have often overlooked the complexities of deploying and managing

containerized blockchain applications, neglecting the issues related to orchestration, integration and efficient resource allocation.

Based on the identified drawbacks and research gaps, the main research direction of this study is to comprehensively investigate the benefits and limitations of containerizing blockchain applications while focusing on improving overall performance. This study has explored the benefits and implications of container-based virtualization in the context of blockchain applications. The findings highlight containers' significant advantages, emphasizing their flexibility, portability and security features.

Using containers, blockchain nodes can be efficiently deployed, managed and isolated, ensuring the integrity and privacy of blockchain networks. The containerization approach enables the seamless movement of applications across different environments and infrastructures, supporting container runtimes and enhancing flexibility and scalability.

Moreover, the study emphasizes that container-based virtualization provides an efficient and lightweight method for running blockchain-based applications and services. Utilizing tools and platforms such as Docker, Kubernetes and AWS Fargate, simplifies the large-scale development, deployment and management of containerized blockchain applications.

In summary, this study contributes to understanding containerization's benefits in the realm of blockchain technology. The findings emphasize the value of container-based virtualization in enabling efficient, lightweight and secure execution of blockchain applications. Moving forward, further research and development in this area should focus on exploring advanced container orchestration techniques, enhancing security measures and addressing the challenges associated with containerization and blockchain technology integration. By doing so, the full potential of container-based virtualization in the blockchain domain can be realized, driving innovation and facilitating the adoption of blockchain technology in various industries.

REFERENCES

[1] Z. Zheng, S. Xie, H. Dai, X. Chen and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus and Future Trends," Proc. of the 2017 IEEE Int. Congress on Big Data (BigData Congress), pp. 557–564, 2017.

[2] Z. Zheng, S. Xie, H.-N. Dai, X. Chen and H. Wang, "Blockchain Challenges and Opportunities: A Survey," Int. J. of Web and Grid Services, vol. 14, no. 4, pp. 352–375, 2018.

[3] A. R. Javed, M. A. Hassan, F. Shahzad, W. Ahmed, S. Singh, T. Baker and T. R. Gadekallu, "Integration of Blockchain Technology and Federated Learning in Vehicular (IoT) Networks: A Comprehensive Survey," Sensors, vol. 22, no. 12, p. 4394, 2022.

[4] N. Naydenov and S. Ruseva, "Cloud Container Service Orchestrated with Kubernetes: A State-of-the- art Technology Review and Application Proposal," Int. J. of Advances in Computer Science and Technology, vol. 12, no. 4, 2023.

[5] A. Bhardwaj and C. R. Krishna, "Virtualization in Cloud Computing: Moving from Hypervisor to Containerization: A Survey," Arabian J. for Science and Eng., vol. 46, no. 9, pp. 8585– 8601, 2021.

[6] C. Chen, Z. Qi, Y. Liu and K. Lei, "Using Virtualization for Blockchain Testing," Proc. of the 2nd Int. Conf. on Smart Computing and Communication (SmartCom 2017), pp. 289–299, Shenzhen, China, December 10-12, 2017.

[7] D. C. Nguyen, P. N. Pathirana, M. Ding and A. Seneviratne, "Integration of Blockchain and Cloud of Things: Architecture, Applications and Challenges," IEEE Communications Surveys & Tutorials, vol. 22, no. 4, pp. 2521–2549, 2020.

[8] K. TaeYoung and K. Hyung-Jong, "Blockchain-based Service Performance Evaluation Method Using Native Cloud Environment," Proc. of the 2020 Int. Conf. on Software Security and Assurance (ICSSA), DOI: 10.1109/ICSSA51305.2020.00016, Altoona, PA, USA, 2020.

[9] G. Ramachandra, M. Iftikhar and F. A. Khan, "A Comprehensive Survey on Security in Cloud Computing," Procedia Computer Science, vol. 110, pp. 465–472, 2017.

[10] E. Casalicchio, "Container Orchestration: A Survey," Proc. of Systems Modeling: Methodologies and Tools, Part of the EAI/Springer Innovations in Comm. and Computing Book Series, pp. 221–235, 2019.

[11] N. G. Bachiega, P. S. Souza, S. M. Bruschi and S. D. R. De Souza, "Container-based Performance Evaluation: A Survey and Challenges," Proc. of the 2018 IEEE Int. Conf. on Cloud Engineering (IC2E), pp. 398–403, Orlando, USA, 2018.

[12] S. Shirinbab, L. Lundberg and E. Casalicchio, "Performance Evaluation of Container and Virtual Machine Running Cassandra Workload," Proc. of the 2017 3rd IEEE Int. Conf. of Cloud Computing Technologies and Applications (CloudTech), pp. 1–8, Rabat, Morocco, 2017.

[13] G. M. Diouf, H. Elbiaze and W. Jaafar, "On Byzantine Fault Tolerance in Multi-master Kubernetes Clusters," Future Generation Computer Systems, vol. 109, pp. 407–419, 2020.
[14] M. A. Marques, C. Miers and M. A. Simplício Jr, "Container Allocation and Deallocation Traceability Using Docker Swarm with Consortium Hyperledger Blockchain," Proc. of the 11th Int. Conf. on Cloud Computing and Services Science, vol. 1: CLOSER, pp. 288–295, 2021.
[15] J. Islam, Container-based Microservice Architecture for Local IoT Services, PhD Thesis, University of Oulu, Oulu, Finland, 2019.
[16] A. Brinckman, D. Luc, J. Nabrzyski et al., "A Comparative Evaluation of Blockchain Systems for Application Sharing Using Containers," Proc. of the 13th IEEE Int. Conf. on e-Science (e-Science), pp. 490–497, Auckland, New Zealand, 2017.
[17] A. S. Alsaffar and A. H. Alezzy, "A Lightweight Portable Multithreaded Client-server Docker Containers," Technium: Romanian J. of Applied Sciences and Technol., vol. 4, no. 10, pp. 31–43, 2022.
[18] W. Li, J. Wu, J. Cao, N. Chen, Q. Zhang and R. Buyya, "Blockchain-based Trust Management in Cloud Computing Systems: A Taxonomy, Review and Future Directions," Journal of Cloud Computing, vol. 10, no. 1, pp. 1–34, 2021.
[19] P. Raj, J. S. Chelladhurai and V. Singh, Learning Docker, ISBN: 1784397938, Packt Publish. Ltd., 2015.
[20] J. Sun, C. Wu and J. Ye, "Blockchain-based Automated Container Cloud Security Enhancement System," Proc. of the IEEE Int. Conf. on Smart Cloud (SmartCloud), pp. 1–6, Washington, USA, 2020.
[21] A. Mouat, Using Docker: Developing and Deploying Software with Containers, ISBN: 9781491915769, O'Reilly Media, Inc., 2015.
[22] D. Yaga, P. Mell, N. Roby and K. Scarfone, "Blockchain Technology Overview," arXiv preprint, arXiv: 1906.11078, 2019.
[23] V. Bakayov and A. Custură, "Blockchain Evolution," Tech. Rep., Research Institute, Amsterdam, Netherlands, 2020.
[24] S. Zeadally and J. B. Abdo, "Blockchain: Trends and Future Opportunities," Internet Technology Letters, vol. 2, no. 6, p. e130, 2019.
[25] S. Lemeš, "Blockchain-based Data Integrity for Collaborative Cad," Proc. of Mixed Reality and Three-dimensional Computer Graphics, IntechOpen, pp. 1–17, 2020.
[26] R. Jabbar, E. Dhib, A. B. Said, M. Krichen, N. Fetais, E. Zaidan and K. Barkaoui, "Blockchain Technology for Intelligent Transportation Systems: A Systematic Literature Review," IEEE Access, vol. 10, pp. 20 995–21 031, 2022.
[27] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman et al., "Blockchain Technology: beyond Bitcoin," Applied Innovation, vol. 2, no. 6-10, p. 71, 2016.
[28] J. C. López-Pimentel, O. Rojas and R. Monroy, "Blockchain and Off-chain: A Solution for Audit Issues in Supply Chain Systems," Proc. of the IEEE Int. Conf. on Blockchain (Blockchain), pp. 126–133, Rhodes, Greece, 2020.
[29] S. Chaisawat and C. Vorakulpipat, "Fault-tolerant Architecture Design for Blockchain-based Electronics Voting System," Proc. of the 17th IEEE Int. Joint Conf. on Computer Science and Software Engineering (JCSSE), pp. 116–121, Bangkok, Thailand, 2020.
[30] Y. Sun, L. Wu, S. Wu, S. Li, T. Zhang, L. Zhang, J. Xu and Y. Xiong, "Security and Privacy in the Internet of Vehicles," Proc. of the IEEE Int. Conf. on Identification, Information and Knowledge in the Internet of Things (IIKI), pp. 116–121, Beijing, China, 2015.
[31] J. Golosova and A. Romanovs, "The Advantages and Disadvantages of the Blockchain Technology," Proc. of the 6th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), pp. 1–6, 2018.
[32] S. Kim and G. C. Deka, Advanced Applications of Blockchain Technology, ISBN: 9789811387753, Springer, 2020.
[33] A. Yewale, Study of Blockchain-as-a-service Systems with a Case Study of Hyperledger Fabric Implementation on Kubernetes, PhD Thesis, University of Nevada, Las Vegas, USA, 2018.
[34] D. Efanov and P. Roschin, "The All-pervasiveness of the Blockchain Technology," Procedia Computer Science, vol. 123, pp. 116–121, 2018.
[35] M. Mazzoni, A. Corradi and V. Di Nicola, "Performance Evaluation of Permissioned Blockchains for Financial Applications: The Consensys Quorum Case Study," Blockchain: Research and Applications, vol. 3, no. 1, p. 100026, 2022.
[36] G. Hileman and M. Rauchs, "2017 Global Blockchain Benchmarking Study," SSRN, no. 3040224, p. 122, 2017.
[37] S. Gec, D. Lavbič, M. Bajec and V. Stankovski, "Smart Contracts for Container Based Video Conferencing Services: Architecture and Implementation," Proc. of the 15th Int. Conf. in Economics of Grids, Clouds, Systems and Services (GECON 2018), pp. 219–233, Pisa, Italy, Springer, 2019.
[38] A. Ahmad, A. Alabduljabbar, M. Saad, D. Nyang, J. Kim and D. Mohaisen, "Empirically Comparing the Performance of Blockchain's Consensus Algorithms," IET Blockchain, vol. 1, no. 1, pp. 56–64, 2021.
[39] X. Wu, J. Yan and D. Jin, "Virtual-time-accelerated Emulation for Blockchain Network and Application

Evaluation," Proc. of the 2019 ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation, pp. 149–160, 2019.
[40] S. Pongnumkul, C. Siripanpornchana and S. Thajchayapong, "Performance Analysis of Private Blockchain Platforms in Varying Workloads," Proc. of the 26th IEEE Int. Conf. on Computer Communication and Networks (ICCCN), pp. 1–6, Vancouver, Canada, 2017.
[41] Y. Hassanzadeh-Nazarabadi, A. Küpçü and Ö. Özkasap, "Lightchain: A DHT-based Blockchain for Resource Constrained Environments," arXiv preprint, arXiv: 1904.00375, 2019.
[42] Z. Shi, C. Jiang, L. Jiang and X. Liu, "HPKS: High Performance Kubernetes Scheduling for Dynamic Blockchain Workloads in Cloud Computing," Proc. of the 14th IEEE Int. Conf. on Cloud Computing (CLOUD), pp. 456–466, Chicago, USA, 2021.
[43] G. Volpe, A. M. Mangini and M. P. Fanti, "An Architecture for Digital Processes in Manufacturing with Blockchain, Docker and Cloud Storage," Proc. of the 17th IEEE Int. Conf. on Automation Science and Engineering (CASE), pp. 39–44, Lyon, France, 2021.
[44] N. El Ioini and C. Pahl, "Trustworthy Orchestration of Container Based Edge Computing Using Permissioned Blockchain," Proc. of the 5th IEEE Int. Conf. on Internet of Things: Systems, Management and Security, pp. 147–154, Valencia, Spain, 2018.
[45] D. Tosh, S. Shetty, P. Foytik, C. Kamhoua and L. Njilla, "CloudPoS: A Proof-of-stake Consensus Design for Blockchain Integrated Cloud," Proc. of the 11th IEEE Int. Conf. on Cloud Computing (CLOUD), pp. 302–309, San Francisco, USA, 2018.
[46] K. Awuson-David, T. Al-Hadhrami, O. Funminiyi and A. Lotfi, "Using Hyperledger Fabric Blockchain to Maintain the Integrity of Digital Evidence in a Containerized Cloud Ecosystem," Proc. of the Int. Conf. of Reliable Information and Communication Technology (IRICT 2019), Emerging Trends in Intelligent Computing and Informatics, pp. 839–848, Springer, 2020.
[47] G. S. Aujla, A. Singh, M. Singh, S. Sharma, N. Kumar and K.-K. R. Choo, "Blocked: Blockchain-based Secure Data Processing Framework in Edge Envisioned v2x Environment," IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 5850–5863, 2020.
[48] P. Kumar and M. Shah, "To Build Scalable and Portable Blockchain Application Using Docker," Proc. of Soft Computing: Theories and Applications (SoCTA 2019), Part of the Advances in Intelligent Systems and Computing Book Series, vol. 1154, pp. 619–628, Springer, 2020.
[49] O. Bentaleb, A. S. Belloum, A. Sebaa and A. El-Maouhab, "Containerization Technologies: Taxonomies, Applications and Challenges," The Journal of Supercomputing, vol. 78, no. 1, pp. 1144–1181, 2022.

ملخص البحث:

حظيت تكنولوجيا سلاسل الكُتل بالكثير من الاهتمام في مجالاتٍ علميةٍ وهندسيةٍ متعدّدة. ولتحسين خدمات تكنولوجيا سلاسل الكُتل، لا بُدَّ من معالجة التّحديات التي تواجه تطبيقها. ويُعدّ استخدام ما يسمى "الأوعية الافتراضية" من الطُّرق التي يمكنها إدخال عددٍ من التّحسينات على خدمات تكنولوجيا سلاسل الكُتل من عدّة جوانب، أبرزها الحجم والتّعقيد والأمان وإمكانية التّوسيع لشبكات تكنولوجيا سلاسل الكُتل التّقليدية.

تشرح هذه الورقة تكنولوجيا سلاسل الكُتل، والأسباب التي وَراء دمْجها مع "الأوعية الافتراضية". ومن ناحية أخرى، فإنّ تكنولوجيا الأوعية الافتراضية تَستخدم سلاسل الكُتل لحماية البيانات، وتحسين إدارة الموارد. كذلك تعمل هذه الدراسة على تحليل الدراسات التي تناولت كُلاً من تكنولوجيا سلاسل الكُتل وتكنولوجيا الأوعية الافتراضية. وترتكز الدراسة على أنّ التّكامل بين تكنولوجيا سلاسل الكُتل والأوعية الافتراضية يتيح وضْع كُلّ تطبيقٍ من تطبيقات سلاسل الكُتل في وعاءٍ افتراضي منفصل، بحيث لا يؤثّر أي خللٍ في أحد الأوعية على الأوعية الأخرى أو على شبكة سلاسل الكُتل برُمّتها.

وفي الختام، تستعرض الدراسة الصّعوبات التي تكتنف التّكامل بين سلاسل الكُتل و الأوعية الافتراضية، وتناقش اتّجاهات البحث المستقبلية من أجْل إدخال تحسيناتٍ على نتائج هذا البحث.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

A BLENDED SOFT-COMPUTING MODEL FOR STOCK-VALUE PREDICTION

N. Usha Devi and R. Mohan

(Received: 13-May-2023, Revised: 13-Jul.-2023, Accepted: 19-Jul.-2023)

ABSTRACT

Stock investments play a crucial role in deciding the global economic growth of the country. Investors can optimize profit and avoid risk through accurate stock-value prediction models, which motivates researchers to work on various aspects of correlated features and predictive models for stock-value prediction. The existing stock-value prediction models used data like Twitter, microblogs, price history and Google trends. On the other hand, domain-specific dictionary-based deep learning evolved as a competitive model for alternative models in stock value prediction. But, the accuracy of these models depends on the quality of the input, the correlation among the features and the correctness of the sentiment scores generated for the dictionary terms. Financial-news sentiment analysis for stock-value prediction with dictionary-based learning needs attention in improving the quality of the input and dictionary terms' sentiment score generation. The present research aims to develop a blended soft-computing model for stock-value prediction (BSCM) with cooperative fusion and dictionary-based deep learning. In the current work, six Indian stocks that cover uptrend, sideways and downtrend characteristics are considered with stock-price histories and news headlines from 8th August 2016 to 31st March 2023, i.e., 2427 days. The number of records in price-history dataset is 14,562 and in the news headlines dataset is 46,213. The performance of the stock-value prediction can be improved by taking advantage of multi-source information and context-aware learning. The present research aims to achieve three objectives: 1. Applying cooperative fusion to combine the news headlines and price history of stocks collected from multiple sources to improve the quality of the input with correlated features. 2. Building a dictionary, FNSentiment, with a novel strategy. 3. Predicting stock values using FNSentiment and News Sentiment Prediction Model (NSPM) integration. In the experimentation, the proposed model outperformed the state-of-the-art models with an accuracy of 91.11%, RMSE of 10.35, MAPE of 0.02 and MAE of 2.74.

KEYWORDS

Deep learning, Stock market, Sentiment analysis, Fusion, Sentiment dictionary.

1. INTRODUCTION

The stock-value prediction models gained attention in recent times. The stock value varies with the variety of features. The major categories of these features are internal and external features. Internal features include close price, the price-to-earnings ratio (P/E), the price-to-book ratio (P/B) and the like [1]. The news, Twitter, Google trends and other social media reveal external features: currency value, political decisions, organization profit, loss, the relation between employees and chief executive officer and the like, as described by [2]. The selection of the features dramatically impacts the stock-value prediction accuracy. The popular stock-value prediction models considered price history, news, Twitter [3] and microblogs as sources of stock information. But, social groups or microblogs cover a limited number of people's opinions that cannot be trusted compared to the news media reports.

In recent research, news financial sentiment analysis models evolved to predict the stock price [4]. An interesting correlation was found between the news features and the stock price. In the proposed model, to join these correlated features, cooperative fusion [5] is used. Cooperative fusion is a methodology to combine correlated features from various sources to improve the quality of the input. Through this fusion methodology, the performance of the predictive model can be improved.

Financial news sentiment analysis involves generating sentiment scores for the words in the input news. The existing English-term dictionaries were insufficient to capture the meaning of the business context statements. The Natural Language Processing (NLP) problem, polysemy [6] means that multiple interpretations are possible for the same word depending on the context. Hence, domain-specific dictionaries are required to improve the accuracy of sentiment prediction.

N. Usha Devi and R. Mohan are with NIT Tiruchirappalli, Tuvakudi, Tamil Nadu, India. Emails: 406915052@nitt.edu and rmohan@nitt.edu

Domain–specific dictionaries for stock value prediction exist for various languages. But, these dictionaries need improvement in considering correlated features of terms to generate accurate sentiment scores in dictionary learning. In the proposed system, the novel dictionary FNSentiment is developed with fused information by combining correlated features, news and close price.

Ahmad et al. [7] expressed that the accuracy of the sentiments depends on the learning model's performance. Deep-learning models exhibit great computation power in fields like image processing and text analytics. In specific, Deep Neural Networks (DNNs) [8], like Convolutional Neural networks (CNNs) [9], Long Short Term Memory (LSTM) [10] and Gated Recurrent units (GRUs) [11] are dynamic classification and predictive models. The proposed model uses the NSPM model, which combines CNN and LSTM to compute the stock value. The objectives of the proposed work are as follows.

1) Developing a cooperative fusion methodology to combine the news headlines and price history to improve the quality of the input data to the stock-value prediction model.
2) Generating domain–based dictionary, FNSentiment to with dictionary-based deep learning using the fused information.
3) Developing a stock-value prediction model combining FNSentiment and NSPM.

The remaining sections of the paper are outlined as follows: Section 2 presents the related work. Section 3 explains the theory and implementations of the proposed model. Section 4, interprets the results and discussion. Section 5 presents the conclusion and some suggestions for future work.

2. RELATED WORK

This section presents the various methods to predict stock prices and the importance of dictionary-based deep learning for stock-value prediction. Further, the section discusses the advantages and limitations and steps to overcome the boundaries of the existing models and give a clear vision of the proposed model.

2.1 Stock-value Prediction

Stock trading is a business investment technique that results in drastic variations in profit or loss in a short span with a quick change in stock value. In recent years, the following methods have evolved to address the stock-market analysis and prediction; Auto Regression (AR), Auto Regressive Integrated Moving Average (ARIMA), Auto Regressive Moving Average (ARMA) models, linear regression, Support Vector Regression (SVR), Bayes neural network, Hybrid Network Adaptive Time-series recommendation framework (HNATS) [12], Long Short Term Memory Cellular Automata (LSTMCA) [13], Fuzzy time series analysis [14], GRU [15]. Stock-value prediction is a time-series problem involving statistical or textual data analysis.

The researchers used various statistical models to address stock-value prediction based on internal features. Kumar et al. [16] have proposed an SVR with the fuzzy model and a genetic algorithm with SVR [17] to perform time-series analysis on statistical data stock's close price. Tunisian stock data analysis using a hierarchical deep neural network [18] showed a considerable performance. Even though the time-series prediction analyzes the stock's close price and other statistical factors like the P/E ratio, this analysis ignored external features that dynamically decide stock variations.

The time-series text analysis involves investigating the stock-market data collected from several media. Long et.al [19] have used the news media analysis for stock-value prediction using SVM with S&S kernel and obtained good performance. In this study, the authors expressed the importance of considering the news data for stock-value prediction. Many researchers have experimented with mass media, news, Twitter, microblogs [20], online financial comments [21], behavioural finance [22] and the like and found that the media creates hype and touches user emotions in stock trading [23]. However, in social-media analysis for stock-value prediction, all the traders must be members of a specific social network with active communication to capture data, which is impossible. In addition, the users might be irrational; hence, the correctness of the social-media data is questionable.

To overcome the existing models' limitations, we considered news data and price history for stock-value prediction in the present study. News is quickly captured and reachable to traders *via* newspapers and electronic media. Moreover, the news data is trustworthy when compared with social-media data. We

have developed a cooperative fusion method to use the price history and news headlines to obtain input datasets for stock-value prediction.

Table 1. The classification of the news headlines into positive, negative and neutral.

Stock	No. of Sentences	The polarity of news sentence		
		Positive	Negative	Neutral
WIPRO	6262	2993	2273	996
TCS	9114	6414	1580	1120
BHARTIARTL	5812	2490	1812	1510
SBIN	12855	7795	3234	1826
NXTDIGITAL	5013	2907	1488	618
PNB	7157	2980	2887	1290

2.2 Dictionary-based Sentiment Analysis

In recent research, many dictionaries have evolved to analyze text belonging to various domains. These dictionaries have been developed to be object-specific, category-specific, language-specific and the like. SentiDomain [24] was introduced as a rule-based sentiment dictionary developed for particular domain objects. This work calculates the sentiments for each object cluster using cosine similarity. This method analyzed the user review to measure product rankings and user satisfaction. Loughran and Mc- Donald manually developed a financial news dictionary, LMFinance for Hong Kong news. This dictionary outperformed the existing dictionaries, SentiWordNet and Senticnet [25].

A Korean-language dictionary [26] was developed to extract nouns from news statements and sentiments of positive and negative words obtained by calculating the average frequency of all positive and negative words in the Korean language. Most of the dictionaries were built in Japanese and Chinese rather than in English. We studied various methods to create dictionaries for deriving a domain-specific dictionary. Then, we analyzed the advantages of manual and automated dictionaries. Consequently, we proposed a semi-automated FNSentiment dictionary to take the benefits of both manual and automated dictionaries.

2.3 Sentiment Analysis with Deep Learning

The challenges in the sentiment analysis made researchers tend towards using dynamic computing models. In recent studies, deep-learning models have shown exemplary performance in sentiment analysis. Abdi et.al. [27] have introduced RNSA, which uses RNN and LSTM combinations for sentiment analysis. This model finds the sentiments of the users' emotions in social-media data. The authors found that in word-level and sentence-level features with pre-trained embedding vectors, Word2vec showed promising results.

Chen et.al. [28] have used GRU to analyze a dictionary created from Chinese social networks. This model classifies the user's emotions as positive or negative. Then, these sentiments are used to predict the stock price. Later, the authors proposed another deep-learning model, RNNboost [29], for stock prediction. The authors believed deep learning efficiently finds user emotions and is reliable for stock-value prediction. In addition, they showed that dictionary-based knowledge is suitable for analyzing domain-specific data. The hybrid CNN-LSTM dynamically classifies the statements positively and negatively [30]. In the present work, we have designed a deep-learning model, News sentiment-prediction model (NSPM) that employs CNN and LSTM with Word2Vec embedding in the proposed system.

3. PROPOSED SYSTEM

This section presents the theory and implementations of blended soft-computing model for stock-value prediction (BSCM) for the stock-value prediction for news updates using a dictionary-based deep learning approach. Figure 1 shows the BSCM architecture. The design and development of BSCM are as follows:

1) The news headlines are collected from the National Stock Exchange (NSE) and Times of India (ToI). Price history is collected from the NSE.
2) The cooperative fusion method combines price history and news headlines information from

multiple sources.

3) The FNSentiment dictionary was developed that consists of significant bigram terms with close price and sentiment scores.
4) NSPM is modeled using the deep-learning approach for stock-value prediction when integrated with FNSentiment.

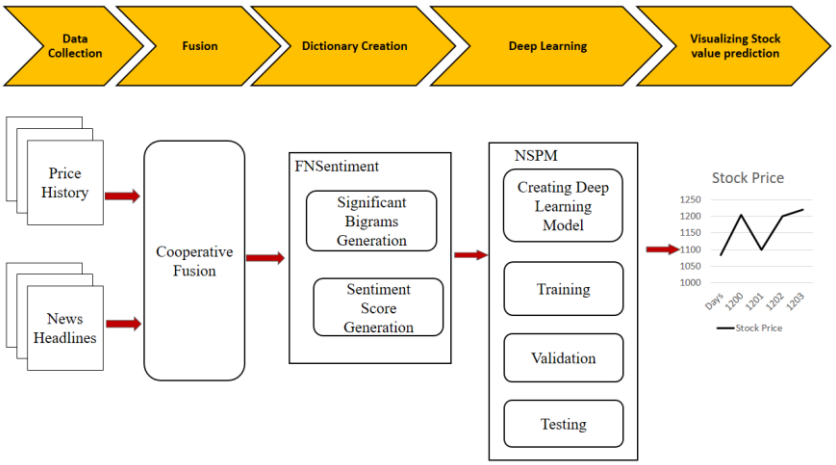

Figure 1. Architecture of the proposed model.

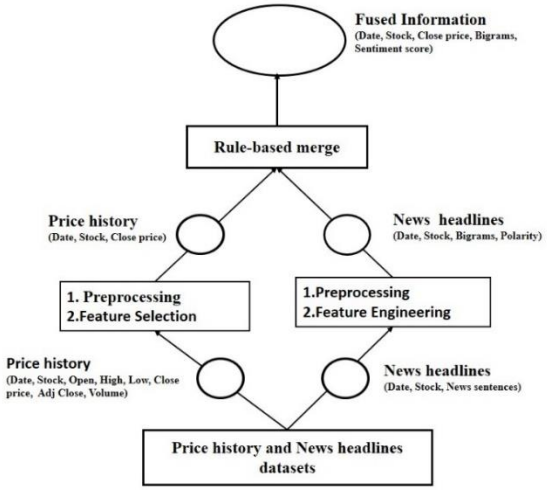

Figure 2. The workflow of cooperative fusion model.

Table 2. Sample terms with sentiment score in FNSentiment dictionary.

Significant bigram	Sentiment score	Close price (R s)
bad debt	0.034	490
crore loss	0.126	517
delays cheque	0.474	693
hits low	0.042	642
clearing debt	0.712	1100
more growth	0.854	1322
profit drops	0.038	786

3.1 Data Collection

The experimentation in this research considers six stocks WIPRO, TCS, BHARIARTL, SBIN, NXTDIG- ITAL and PNB to cover three possible trends in the stocks. Figure 4 illustrates the price-history input datasets with three dataset characteristics; upward, sideways and down trends. The stock datasets are of two types; one is price-history data and the other one is news headlines data from 8th August 2016 to 31st March 2023. The price-history data represents day to day transactions of trading.

The data has been collected from the NSE. The dataset contains everyday transaction details: Date, Adjacent close price, High, Low, Close price and Volume. The dataset size for each stock is 2427 for the trading days from 8th August 2016 to 31st March 2023. The total size of the price-history data is 14,562 for the six stocks. The news headlines dataset is framed by extracting data from two business news sources for the six stocks: 1. ToI, India's primary new media in English and 2. NSE press-release descriptions. The news database contains 46,213 sentences. This dataset consists of news headlines and the date of the news dissemination. Now the datasets are ready for fusion to obtain the quality inputs for stock-value prediction.

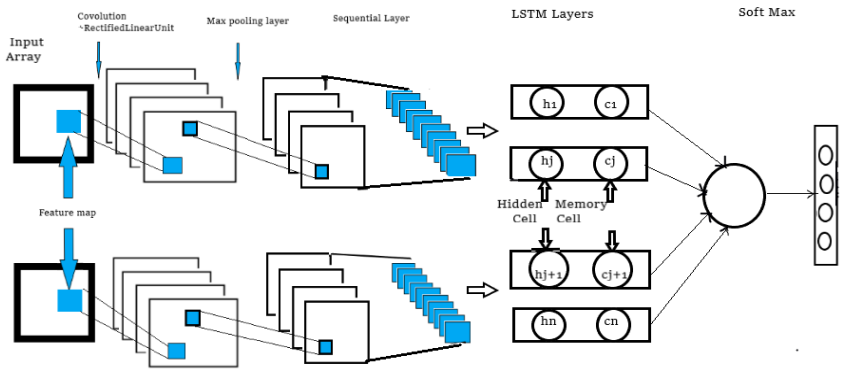

Figure 3. The layered architecture of news sentiment prediction model.

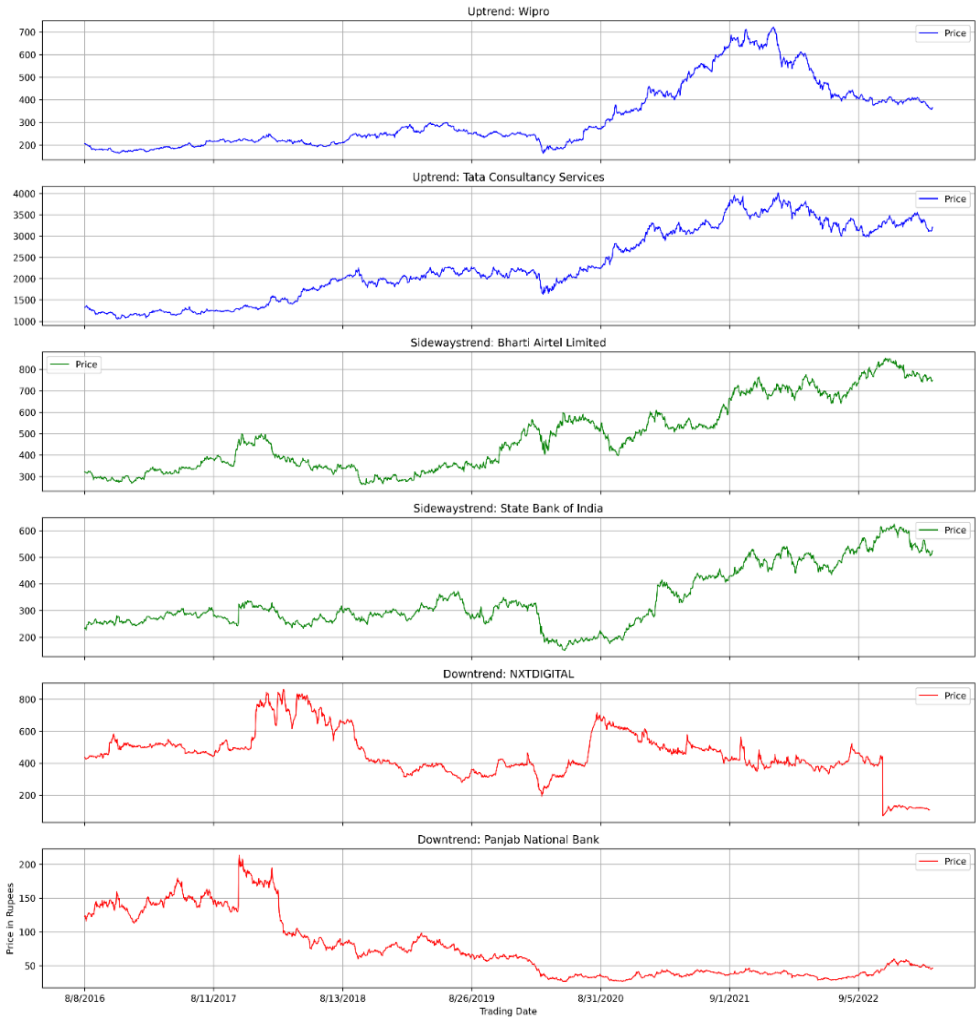

Figure 4. The stock input datasets with three characteristics; uptrend, sideways and down trends.

3.2 Fusion Method

The input datasets to build the dictionary were obtained by cooperative fusion, as illustrated in Figure 2. After data collection, the pre-processing and feature-selection methods were used to find the essential elements of the price-history data. In the pre-processing step, the null value of data will be substituted by the previous day's instance value. The fusion method was applied to the datasets as follows:

1) Price-history pre-processing and feature selection.

a) Pre-processing: The stop words and the like were eliminated from the news sentences.

b) Feature engineering: For each sentence in the news headlines:

(i) A sequence of bigrams (two consecutive words) from the pre-processed news sentence is generated with a semicolon as a separator. In the next step, a polarity feature ranging from -1 to 1 will be added to each headline entry based on domain knowledge. Table 1 summarizes the domain knowledge about the news headlines.

(ii) Finally, the bigrams with the same date and stock entries are joined with semicolon as a separator and the new polarity value is considered as the sum of polarities in the respective entries.

2) In the rule-based merge step, the close-price data is mapped with news headlines data and *vice versa* based on date and stock features.

a) For the available entry in news headlines data, if the corresponding date entry is missing in the price history, a new entry is created with date and close price by considering the 1 to n steps available close price for the stock.

b) If the corresponding date entry is missing in the news headlines data, then move back to the 1 to n steps to find the most recent news for the stock.

c) For common date and stock, the stock's price-history is merged with the news headlines data.

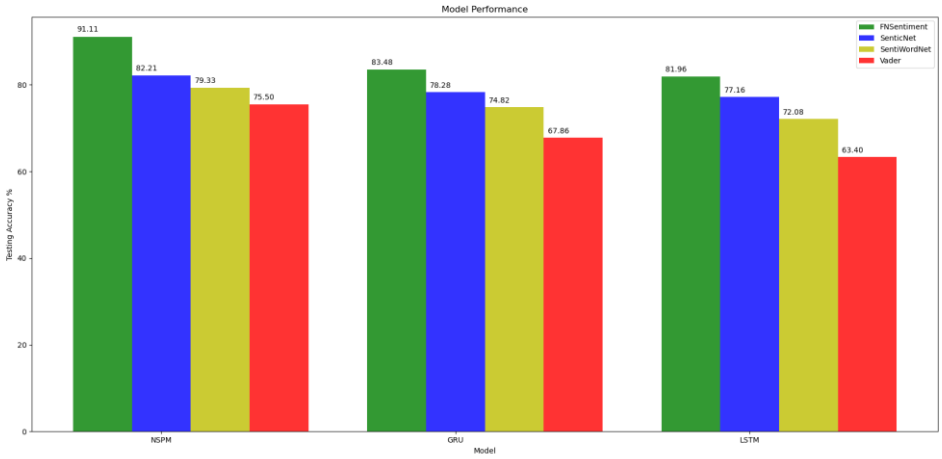

Figure 5. Performance of the news sentiment-prediction model with various standard dictionaries.

$$S_i = \frac{s}{(n-1)} \tag{1}$$

$$ST_i = \sum S_j \ for\ j = 1\ to\ k \tag{2}$$

$$NST_i = \frac{ST_i - Min}{Max - Min} \tag{3}$$

3.3 Financial News Term Sentiment Dictionary (FNSentiment)

The fabrication of FNSentiment consists of two steps 1. Computing significant bigrams, 2. Computing bigram polarities. In the first step, significant bigram generation produces the critical terms for news sentiment analysis. The significant bigrams were obtained by the two consecutive terms generated by combining a noun, adjective, adverb and verb by discarding the other terms in each news sentence. The

polarity computation finds the polarity values for all sentences. The sentence with word length n+1 and score S generates n terms. The S gives the sentiment score for each significant term in the sentence. A sentiment score S_i for i=1 to n was computed using (1), contributing to the sentiment score; hence, the output was a set of terms with sentiment for each sentence. Next, the sentence's sentiment denoted by ST*i* was computed using (2). After this step, The FNSentiment was updated with the pairs of terms and their scores. Next, the Normalized Sentiment Score (NST) was computed using (3) to convert the sentiment scores on the scale (0, 1). Thus, NST*i* generates polarity values ranging from (0, 1). The NST*i* value zero represents the highly negative sentiment term. The NST*i* value one defines the extreme positive sentiment term and 0.5 means the terms with a neutral sentiment. In (3), min and max represent ST's minimum and maximum values, respectively. Table 3 shows an instance of the FNSentiment dictionary.

3.4 Stock-value Prediction Using NSPM

The objective of the NSPM is to predict the news sentiments using deep learning, as shown in Figure 3. The model consists of three layers; embedding, CNN and LSTM. First, the embedding layer creates input vectors to train the model. In this step, the layer generates equivalent word vectors for the inputs. In the next step, the embedded vectors were passed from the convolution layer to the max pooling layer to capture the most significant features from the information. Finally, the input was passed through the sequential layers of LSTM for further learning. The ci indicates an internal cell that collects input in LSTM. The hj represents the hidden state that produces output. The final layer softmax outputs a value from zero to one, indicating the sentence sentiment value. The hyper-parameters of NSPM were found through the Bayesian optimization tuner of the Keras tuner.

Table 3. An instance of fused information obtained from the news and price-history datasets.

Date	**Stock**	**Close price (R s)**	**Significant bigrams**	**Sentiment score**
17-Apr-21	SBIN	339.9	statebankofindia private; private bank; bank seen; seen race; race card; card business	1
18-Apr-21	SBIN	339.9	statebankofindia pharmacist; pharmacist recruitment apply; apply online	0
20-Apr-21	SBIN	329.5	statebankofindia say; say charge; charge zero; zero balance; balance account; account prior; prior reasonable	1
24-Apr-21	SBIN	336.45	statebankofindia cut; cut growth; growth vijay; vijay mallya; mallya say; say money; money owes; owes indian; indian bank; bank public; public money; money cannot; cannot made; made bankrupt	-1
25-Apr-21	SBIN	336.45	bihar suffer; suffer crore; crore financial; financial loss; loss covid	-1
27-Apr-21	SBIN	353.05	statebankofindia clerk; clerk exam; exam registration; registration begin	0

Table 4. Significant bigram and sentiment-score generation for the sample fused information to develop FNSentiment.

Significant bigrams	**Sentiment score**	**Normalized sentiment score**	**Close price**
statebankofindia private	0.17	1	339.9
private bank	0.17	1	339.9
bank seen	0.17	1	339.9
seen race	0.17	1	339.9
race card	0.17	1	339.9
card business	0.17	1	339.9

statebankofindia pharmacist	0	0.54	339.9
pharmacist recruitment	0	0.54	339.9
recruitment apply	0	0.54	339.9
apply online	0	0.54	339.9
bihar suffer	-0.2	0	336.4
suffer crore	-0.2	0	336.4
crore financial	-0.2	0	336.4
financial loss	-0.2	0	336.4
loss covid	-0.2	0	336.4

Table 5. Performance comparison of news sentiment-prediction model with baseline models.

Model	**Phase**	**Accuracy (%)**						**Avg.**
		WIPRO	**TCS**	**BHARTI**	**SBIN**	**NXT**	**PNB**	
NSPM with FNSentiment	Testing	91.33	92.51	90.35	91.06	90.14	91.25	91.11
	Training	92.32	93.63	91.08	93.14	91.72	92.19	92.35
	Validation	89.45	89.87	88.66	88.91	88.32	89.52	89.12
NSPM	Testing	86.11	84.34	83.47	85.17	83.79	80.14	83.84
	Training	88.99	86.44	88.67	87.91	86.90	82.39	86.88
	Validation	85.59	83.44	82.15	84.10	82.84	79.49	82.94
LSTM With FNSentiment	Testing	86.62	85.14	85.38	82.43	81.06	80.24	83.48
	Training	87.99	87.74	86.55	83.93	82.41	81.32	84.99
	Validation	84.10	83.56	83.15	80.21	80.41	79.49	81.82
LSTM	Testing	82.12	81.74	80.11	78.26	78.79	78.18	79.87
	Training	85.16	83.45	81.16	81.06	81.16	80.12	82.02
	Validation	80.28	80.14	79.36	75.13	77.54	75.52	78.00
GRU With FNSentiment	Testing	85.12	81.24	84.18	80.17	81.79	79.24	81.96
	Training	86.56	84.71	85.27	82.06	82.16	80.59	83.56
	Validation	85.59	83.44	82.15	84.10	82.84	79.49	82.94
GRU	Testing	81.84	81.24	83.28	78.18	72.93	71.33	78.13
	Training	84.27	83.82	84.17	80.04	76.62	74.59	80.59
	Validation	79.39	80.29	80.11	75.86	69.93	70.84	76.07

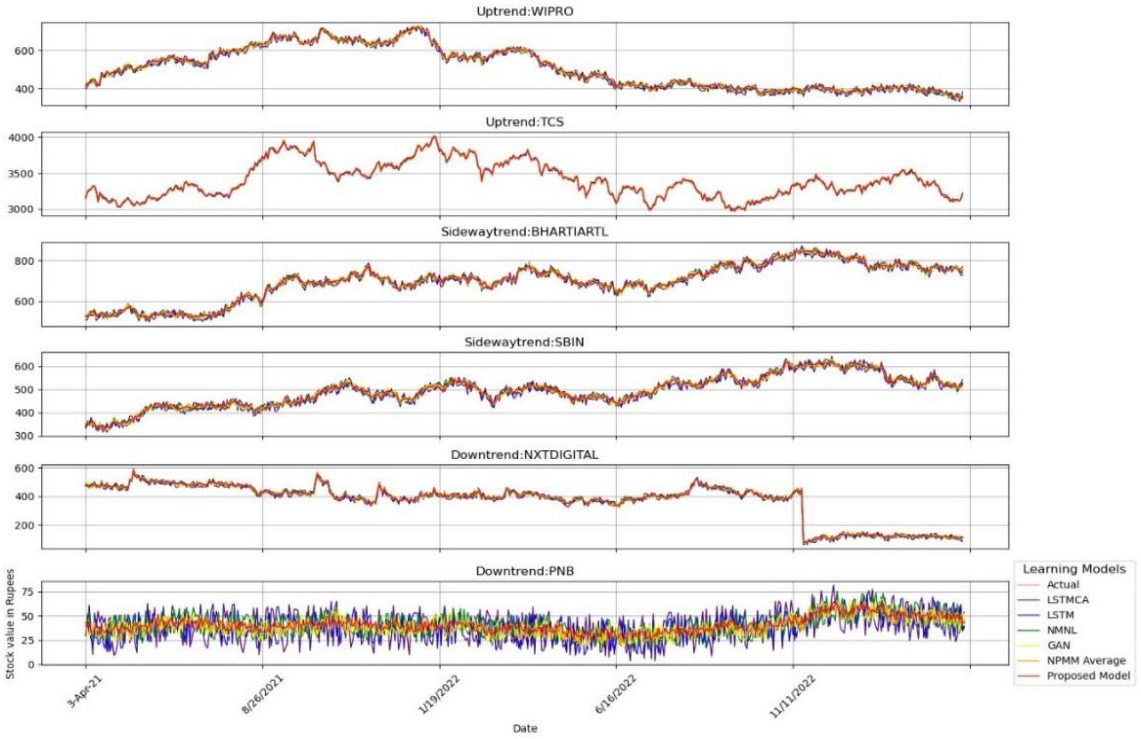

Figure 6. Stock-value prediction using the proposed model and existing models.

4. RESULTS AND DISCUSSION

The dataset is divided into 70% for training data from 8th Aug 2016 to 3rd March 2021 (1699 days) and 30% testing data from 4th March 2021 to 31st March 2023 (728 days). The results, performance analysis of the proposed model, evaluation and comparison with the models in recent literature are explained in the following sub-sections.

Table 6. Comparison of proposed and baseline models using the RMSE metric for each input stock dataset.

Model/ Stock	WIPRO	TCS	BHARTIARTL	SBIN	NXT DIGITA L	PNB
Proposed Model	9.53	10.00	10.88	10.86	10.24	10.58
NPMM Average[31]	24.36	24.26	27.84	24.74	29.44	26.02
GAN [32]	51.23	51.06	47.11	50.34	49.04	50.72
NMNL[33]	76.70	83.83	84.46	86.73	90.42	89.07
LSTM [11]	125.24	127.87	110.64	127.58	125.27	123.01
LSTMCA [13]	200.36	209.52	219.52	218.89	224.35	221.20

Table 7. Comparison of proposed and baseline models using the MAE metric for each input stock dataset.

Model/ Stock	WIPRO	TCS	BHARTIART L	SBIN	N XT DIGITAL	PNB
Proposed Model	2.54	2.70	2.84	2.80	2.76	2.78
NPMM Average[31]	4.25	4.22	4.56	4.32	4.66	4.34
GAN [32]	6.22	6.12	5.92	6.09	6.10	6.19
NMNL[33]	7.51	8.03	8.00	8.05	8.32	8.27
LSTM [11]	9.58	9.73	8.85	9.74	9.73	9.62
LSTMCA [13]	12.05	12.53	12.78	12.86	12.87	12.89

Table 8. Comparison of proposed and baseline models using the MAPE metric for each input stock dataset.

Model/ Stock	WIPRO	TCS	BHARTIARTL	SBIN	N XT DIGITAL	PNB
Proposed Model	0.005	0.001	0.004	0.006	0.010	0.071
NPMM Average[31]	0.009	0.001	0.007	0.009	0.016	0.110
GAN [32]	0.013	0.002	0.009	0.013	0.022	0.159
NMNL[33]	0.016	0.002	0.012	0.017	0.029	0.210
LSTM [11]	0.020	0.003	0.013	0.020	0.034	0.249
LSTMCA [13]	0.025	0.004	0.019	0.027	0.044	0.331

Table 9. Comparison of proposed and baseline models using Accuracy, RMSE, MAE and MAPE.

Model/ Metric	Accuracy (%)	RMSE	MAE	MAPE
Proposed Model	91.11	10.35	2.74	0.02
NPMM Average[31]	86.57	26.11	4.39	0.03
GAN [32]	84.4	49.92	6.11	0.04
NMNL[33]	81.89	85.20	8.03	0.05
LSTM [11]	80.77	123.27	9.54	0.06
LSTMCA [13]	78.16	215.64	12.66	0.07

The experimentation of the present research is described as follows. The collected experimental data, news headlines and close prices, was initially joined using cooperative fusion to generate quality input. In the next step, the FNSentiment dictionary was built by computing significant bigrams and their polarities. As a final step, NSPM is used to predict the stock value with the FNSentiment using dictionary–based deep learning.

4.1 Metrics for Evaluation

The metrics used for model evaluation are Accuracy, RMSE, MAE and MAPE. The accuracy determines the percentage of the number of sentences recognized correctly among the total tested

sentences. The RMSE gives the square root of averaged squared error. The error represents the difference between the actual and predicted values. The MAE gives the absolute difference between the predicted and actual values. MAPE is the mean absolute percentage error that determines the relative error. The proposed work aims to optimize these metrics.

$$Accuracy = \frac{Number\ of\ correctly\ categorized\ news\ sentences}{Total\ number\ of\ sentences} \tag{4}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Actual_i - Predicted_i)}{N}} \tag{5}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}(Actual_i - Predicted_i) \tag{6}$$

$$MAPE = \sum_{i=1}^{N}\left|\frac{(Actual_i - Predicted_i)}{Actual_i}\right| \tag{7}$$

4.2 Results of Cooperative Fusion

The news and price-history datasets were combined to create the fused information. In this process, feature selection is applied to price-history datasets to select the prominent features 'close price' and 'Date'. Feature engineering is applied to pre-processed news data sentences to obtain bigrams. Further, a sentiment score between -1 and 1 is appended to each news sentence based on the domain knowledge. In the next step, the close price and bigrams of specific stock were mapped with Date. This step results in datasets with (Date, Stock, Close price, Bigrams, Sentiment score). Table 3 shows information fusion for the sample input dataset.

4.3 FNSentiment Dictionary

The formulation of the FNSentiment dictionary starts by collecting the fused information from the previous step. The next step generates significant bigrams by considering the noun, adjective, adverb and verb combinations of the bigrams obtained from the fused information. In the next step, formulae (1) and (2) are applied to compute the sentiment score for each bigram. Then, normalized sentiment score is calculated for the sentiment score field to convert the range of the values from 0 to 1. For the duplicate terms in the dictionary, sentiment scores were summed up and the close prices were averaged. Now, the FNSentiment contains triplets (significant bigrams, sentiment score, close price). An instance of these results is shown in Table 4.

4.4 Performance of News-sentiment Prediction Model (NSPM)

The NSPM and alternate deep-learning models are integrated with various dictionaries and compared for the analysis. The hyper-parameters of NSPM are done through Bayesian optimization. The hyper-parameters are as follows: The learning rate for the generator and discriminator is 0.01, the suitable optimizer is Adam. We considered the number of epochs as 100 throughout the experimentation. The NSPM with FNSentiment was evaluated and compared with standard dictionaries, SenticNet, SentiWordNet and Vader. Figure 5 illustrates the comparison results. These results demonstrate that context-aware learning is possible through the integration of FNSentiment and NSPM with an accuracy of 91.11%.

Table 5 shows the experiment summary on the six stocks. The NSPM with FNSentiment is a promising approach compared with NSPM alone, with improved accuracy by 3.33% from the experimental results. Figure 5 illustrates the accuracy of NSPM integrating with the FNSentiment and existing dictionaries. The FNSentiment with NSPM shows an accuracy of 91.11%; thus the results concluded that the NSPM with FNSentiment outperformed the recent literature models.

4.5 Performance Analysis of Existing and Proposed Models

BSCM is evaluated and compared with the baseline models with the metrics: Accuracy, RMSE, MAE and MAPE. The metrics were computed using formulae (5), (6) and (7). Tables 6, 7 and 8 illustrate the model evaluation results using the metrics. The summary of the results is shown in Table 9. The BSCM model outperformed all the baseline models with an accuracy of 91.11%. The evaluation of other metrics, RMSE of 10.35, MAPE of 0.002 and MAE of 2.74, showed that the BSCM is a reliable stock-

value prediction system. Figure 6 shows the stock values predicted by the proposed model BSCM and the existing models for the six stocks: WIPRO, TCS, BHARIARTL, SBIN, NXTDIGITAL and PNB.

5. CONCLUSION AND FUTURE WORK

In the present work, we computed the futuristic stock values for six stock datasets with the proposed model and the models in the literature. The proposed model achieved an accuracy of 91.11%, RMSE of 10.35, MAPE of 0.002 and MAE of 2.74. BSCM improved stock-value prediction with a rise in accuracy by 4.54% and with a fall of the MAE by 1.65, MAPE by 0.01 and RMSE by 15.76 compared with the existing models. BSCM outperformed the models in the literature. The NSPM for news sentiment prediction improved accuracy by 3.72% after integrating with a novel dictionary FNSentiment. The results showed that the cooperative fusion method and dictionary-based deep learning models improved the stock-value prediction accuracy. In future studies, we want to incorporate context-based clustering to refine the significant bigrams in predicting the stock value. The BSCM can be enhanced by integrating with clustering to analyze the critical news features for various stocks like oil, bank, software stocks and the like to develop sector-wise dictionaries to optimize the runtime.

REFERENCES

[1] S. Mittal and C. Nagpal, "Predicting a Reliable Stock for Mid and Long Term Investment," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 10, pp. 8440–8448, 2022.

[2] Q. Li, Y. Chen, J. Wang et al., "Web Media and Stock Markets: A Survey and Future Directions from a Big Data Perspective," IEEE Trans. Knowl. Data Eng., vol. 30, no. 2, pp. 381–399, 2018.

[3] Z. Berradi, M. Lazaar, O. Mahboub, H. Berradi and H. Omara, "Combination of Deep-learning Models to Forecast Stock Price of AAPL and TSLA," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 8, no. 4, pp. 345 – 356, 2022.

[4] S. García-Méndez et al., "Automatic Detection of Relevant Information, Predictions and Forecasts in Financial News through Topic Modeling with Latent Dirichlet Allocation," Applied Intelligence, vol. 2023, pp. 1–19, 2023.

[5] R. A. Mokhtar, R. A. Saeed and H. Alhumyani, "Cooperative Fusion Architecture-based Distributed Spectrum Sensing under Rayleigh Fading Channel," Wireless Personal Communications, vol. 124, no. 1, pp. 839–865, 2022.

[6] A. Moldovan, "Descriptions and Tests for Polysemy," Axiomathes, vol. 31, pp. 229–249, 2021.

[7] S. Ahmad, S. Mishra, F. J. Zareen and S. Jabin, "Sensor-enabled Biometric Signature-based Authentication Method for Smartphone Users," Int. J. of Biometrics, vol. 15, no. 2, pp. 212–232, 2023.

[8] P. Mohanty, J. P. Sahoo and A. K. Nayak, "Application of Deep Learning Approach for Recognition of Voiced Odia Digits," Int. J. of Computational Science and Engineering, vol. 25, no. 5, pp. 513–522, 2022.

[9] Y. Guo, S. Han, C. Shen, Y. Li, X. Yin and Y. Bai, "An Adaptive SVR for High-frequency Stock Price Forecasting," IEEE Access, vol. 6, pp. 11397–11404, 2018.

[10] W. Fang, T. Jiang, K. Jiang et al., "A Method of Automatic Text Summarization Based on Long Short-term Memory," Int. J. of Computational Science and Engineering, vol. 22, no. 1, pp. 39–49, 2020.

[11] S. Yang, "A Novel Study on Deep Learning Framework to Predict and Analyze the Financial Time Series Information," Future Generation Computer Systems, vol. 125, pp. 812–819, 2021.

[12] W. Wang, Y. Shi and R. Luo, "Sparse Representation-based Approach to Prediction for Economic Time Series," IEEE Access, vol. 7, pp. 20614–20618, 2019.

[13] N. Usha Devi and R. Mohan, "Long Short-term Memory with Cellular Automata (LSTMCA) for Stock Value Prediction," Proc. of 3rd Int. Conf. ICDECT-2K19, pp. 841–848, DOI: 10.1007/978-981-15-1097-7_70, 2020.

[14] Y. Wang and L. Han, "Adaptive Time Series Prediction and Recommendation," Information Processing & Management, vol. 58, no. 3, p. 102494, 2021.

[15] J. Hu, Q. Chang and S. Yan, "A GRU-based Hybrid Global Stock Price Index Forecasting Model with Group Decision-making," Int. J. of Computational Science and Eng., vol. 26, no. 1, pp. 12–19, 2023.

[16] G. Kumar, S. Jain and U. P. Singh, "Stock Market Forecasting Using Computational Intelligence: A Survey," Archives of Computational Methods in Eng., vol. 28, pp. 1069-1101, 2020.

[17] F. Iqbal, J. M. Hashmi, B. C. M. Fung et al., "A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction," IEEE Access, vol. 7, pp. 14 637–14 652, 2019.

[18] O. Lachiheb and M. S. Gouider, "A Hierarchical Deep Neural Network Design for Stock Returns Prediction," Procedia Computer Science, vol. 126, pp. 264–272, 2018.

[19] W. Long, L. Song and Y. Tian, "A New Graphic Kernel Method of Stock Price Trend Prediction Based on Financial News Semantic and Structural Similarity," Expert Systems with Applications, vol. 118, pp. 411–424, 2019.

[20] Y. Chang, S. Li, A. Liu and J. Jin, "Quality Assessment of Screen Content Images Based on Multi-stage Dictionary Learning," J. of Visual Comm. and Image Representation, vol. 79, no. July, p. 103248, 2021.
[21] Y. Qian, Z. Li and H. Yuan, "On Exploring the Impact of Users' Bullish-bearish Tendencies in Online Community on the Stock Market," Infor. Processing & Management, vol. 57, no. 5, p. 102209, 2020.
[22] Y. Ge et al., "Beyond Negative and Positive: Exploring the Effects of Emotions in Social Media during the Stock Market Crash," Information Processing & Management, vol. 57, no. 4, p. 102218, 2020.
[23] D. Cabrera, C. Cubillos, A. Cubillos et al., "Affective Algorithm for Controlling Emotional Fluctuation of Artificial Investors in Stock Markets," IEEE Access, vol. 6, pp. 7610–7624, 2018.
[24] M. Ahmed, Q. Chen and Z. Li, "Constructing Domain-dependent Sentiment Dictionary for Sentiment Analysis," Neural Computing and Applications, vol. 32, pp. 14719–14732, 2020.
[25] X. Li, P. Wu and W. Wang, "Incorporating Stock Prices and News Sentiments for Stock Market Prediction: A Case of Hong Kong," Infor. Processing & Management, vol. 57, no. 5, p. 102212, 2020.
[26] J. Kim et al., "Stock Price Prediction through the Sentimental Analysis of News Articles," Proc. of the IEEE 11th Int. Conf. on Ubiquitous and Future Networks (ICUFN), pp. 700–702, Zagreb, Croatia, 2019.
[27] A. Abdi, M. Rahmati and M. M. Ebadzadeh, "Entropy Based Dictionary Learning for Image Classification," Pattern Recognition, vol. 110, p. 107634, 2021.
[28] W. Chen, Y. Zhang et al., "Stock Market Prediction Using Neural Network through News on Online Social Networks," Proc. of the 2017 IEEE Int. Smart Cities Conf. (ISC2), pp. 1–6, Wuxi, China, 2017.
[29] W. Chen, C. K. Yeo, C. T. Lau and B. S. Lee, "Leveraging Social Media News to Predict Stock Index Movement Using RNN-boost," Data & Knowledge Engineering, vol. 118, pp. 14–24, Nov. 2018.
[30] P. K. Jain, V. Saravanan and R. Pamula, "A Hybrid CNN-LSTM: A Deep Learning Approach for Consumer Sentiment Analysis Using Qualitative User-generated Contents," Transactions on Asian and Low-resource Language Information Processing, vol. 20, no. 5, pp. 1–15, 2021.
[31] Y. Han, J. Kim and D. Enke, "A Machine Learning Trading System for the Stock Market Based on N-period Min-max Labeling Using XGBoost," Expert Systems with Applications, vol. 211, p. 118581, 2023.
[32] A. Kumar, A. Alsadoon, P. Prasad et al., "Generative Adversarial Network (GAN) and Enhanced Root Mean Square Error (ERMSE): Deep Learning for Stock Price Movement Prediction," Multimedia Tools and Applications, vol. 81, pp. 3995–4013, 2022.
[33] X. Chen, X. Ma, H. Wang, X. Li and C. Zhang, "A Hierarchical Attention Network for Stock Prediction Based on Attentive Multi-view News Learning," Neurocomputing, vol. 504, pp. 1–15, 2022.

ملخص البحث:

تلعب الاستثمارات في الأسهم دوْراً حاسماً في النّموّ الاقتصادي للدّول ويمكن للمستثمرين تعظيم أرباحهم وتجنّب المخاطرة لدى استخدامهم نماذج توقُّع قيم الأسهم، الأمر الذي حدا بالباحثين للعمل على تطوير نماذج لتوقُّع أسعار الأسهم. وقد ظهر التّعلُّم العميق المرتكز على استخدام القواميس الخاصّة بمجالات معينة كتقنية منافسة بديلة لتوقّع أسعار الأسهم في الأسواق المالية. إلّا أنّ دِقّة النّماذج المستخدمة لهذا الغرض تعتمد على جودة المدخلات، والارتباط بين السِّمات، وصحّة المشاعر المتولّدة من المصطلحات القاموسية. من هُنا فإنّ تحليل البيانات المالية من أجْل توقُّع أسعار الأسهم باستخدام التّعلُّم القائم على القواميس يجب أن يُركّز على تحسين جودة المدخلات وتوليد نتائج عالية بالمشاعر المتضمَّنة في المصطلحات القاموسية.

يهدف هذا البحث الى تطوير نموذج مختلط للحوسبة النّاعمة لتوقُّع قيمة الأسهم باستخدام ما يُعرف بالاندماج المقارن والتّعلّم العميق المستند الى القواميس. ويدرس النّموذج المقترح (6) أسواق للأسهم من حيث تواريخ أسعار الأسهم وعناوين الأخبار في تلك الأسواق. والجدير بالذّكر أنّ الأداء المتعلّق بتوقّع أسعار الأسهم في النّماذج المستخدمة يمكن تحسينُهُ بالاستفادة من مزايا المعلومات متعدّدة المصادر والتّعلُّم الواعي للسّياق. ولدى تجريب النّموذج المقترح، تفوّق على مثيلاته من التّقنيات الواردة في أدبيّات الموضوع؛ فقد حقّق دِقّة بلغت 91.11 % الى جانب تميُّزه في مقاييس التقييم الأخرى (RMSE=10.35؛ MAPE=0.02؛ MAE=2.74).

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 03, September 2023.

TRENDS AND CHALLENGES OF ARABIC CHATBOTS: LITERATURE REVIEW

Yassine Saoudi[1] and Mohamed Mohsen Gammoudi[2]

(Received: 29-May-2023, Revised: 27-Jul.-2023 and 19-Aug.-2023, Accepted: 20-Aug.-2023)

ABSTRACT

Conversational systems have recently garnered increased attention due to advancements in Large Language Models (LLMs) and Language Models for Dialogue Applications (LaMDA). However, conversational Artificial Intelligence (AI) research focuses primarily on English. Despite Arabic being one of the most widely used languages on the Internet, only a few studies have concentrated on Arabic conversational dialogue systems thus far. This study presents a comprehensive qualitative analysis of critical research works in this domain to examine the strengths and limitations of existing approaches. The analysis begins with an overview of chatbot history and classification, then explores the language challenges encountered when developing Generative Arabic Conversational AI. Rule-based/Retrieval-based and deep learning-based approaches for Arabic chatbots are also examined. Furthermore, the study investigates the evolution of Generative Conversational AI with the advancements in deep-learning techniques. It also comprehensively reviews various metrics used to assess conversational systems.

KEYWORDS

Chatbot, LLMs, Arabic conversational AI challenges, Generative artificial intelligence, Arabic question answering systems, Taxonomy, Performance evaluation.

1. INTRODUCTION

Conversational agents, commonly known as chatbots, have become integral to our daily lives. They serve as personal assistants on mobile phones, salespeople on e-commerce sites [1] and healthcare assistants [2], engaging in consistent conversations with humans using natural language in text or voice format. While chatbots have been around for decades, recent advancements in artificial intelligence, particularly in human-language processing, have created more efficient, faster and more powerful bots [1], [3]. The field of conversational systems has gained significant attention, driven by the development of Large Language Models (LLMs) and Language Models for Dialogue Applications (LaMDA). However, it is essential to note that conversational systems encompass a broader range of technologies and approaches beyond LLMs and LaMDA.

Chatbots, considered to have specific goals, can be used in many domains, such as in education [4]-[11] and in healthcare [12]-[15]. According to [11], [16]-[18], chatbots reduce the response time to questions; improve customer service; order products online (Alexa from Amazon [19]); research information; guide the user Rahhal [20] (helping tourists at Saudi Arabia) and assist in flight booking [21] (airline ticket booking). Also, solving technical challenges (1+2) is the key to a successful chatbot as: (1) The relevance of the answer: understanding the user's need and not just words and grammar, (2) The structure of the answer: the development of a conversational structure for the user to be comfortable. The dimensions of speed and efficiency appear several times: chatbots must be fast and efficient [22]. They are the most convenient way to deal with consumers in a timely and satisfying way [17].

Based on our extensive reading and research, we highly recommend clearly defining a chatbot: A chatbot is a software application, with or without an avatar, specifically designed to enable conversations using natural language. It utilizes various idioms to serve a specific purpose, aiming to deliver precise information and create a user experience that closely resembles interacting with real individuals regarding efficiency, speed and effectiveness.

Generative Artificial Intelligence (AI) has made remarkable advancements in revolutionizing our lifestyle, work dynamics and interactions with technology. One area that has recently seen significant progress is the development of Large Language Models (LLMs). One prominent example is the

1. Y. Saoudi is with University of Tunis El Manar, Tunis, Tunisia. Email: yassine.saoudi@fst.utm.tn

2. M. M. Gammoudi is with ISAM Manouba University, Tunisia. Email: gammoudimomo@gmail.com

Generative Pretrained Transformer (GPT) family, which includes GPT-1 developed by OpenAI in 2018 [23], GPT-2 in 2019 [24], GPT-3 in 2020 [25] and the latest addition, GPT-4 in 2023 [26]. Another notable model is BERT (Bidirectional Encoder Representations from Transformers), introduced by researchers at Google AI Language in 2018 [27]. In addition, LaMDA, a Language Model for Dialogue Applications, was introduced in 2022 [28]. These models have showcased their success in tasks, such as question-answering, text summarization, sentiment analysis and named entity recognition.

This paper presents a comprehensive review of Arabic conversational dialogue system research to identify critical gaps in the existing literature and propose future research directions. Unlike previous surveys, our review encompasses recent studies that address all aspects of the chatbot workflow. To ensure the logical coherence of the paper, we have established a clear roadmap that guides the organization of different sections. Additionally, our review investigates the progression of Arabic chatbot development with advancements in deep-learning techniques. Finally, unlike previous surveys that primarily focused on classification and reviewing existing chatbot systems, we go beyond that by delving into the evolution of deep-learning techniques, the challenges specific to Arabic language and the evaluation metrics for assessing Arabic conversational dialogue systems.

The rest of this paper is organized as follows: Section 2 outlines the methodology employed in conducting this study. Section 3 provides a synthetic background as well as chatbot classification and explores some applications of chatbots. Section 4 examines the challenges encountered in Arabic Conversational AI systems. Subsequently, Section 5 explores rule-based and retrieval-based Arabic chatbots. Section 6 focuses on deep learning-based Arabic Question Answering Systems. Progressing further, Section 7 investigates the evolutionary advancements of deep-learning techniques. The assessment metrics for conversational AI and Question-Answering systems are scrutinized in Section 8. Section 9 draws this survey to a close by offering a discussion of the research findings, while Section 10 presents the conclusions and highlights potential horizons future research.

2. METHODOLOGY

This survey examines various approaches employed in conversational dialogue systems and investigates the achievements and obstacles associated with building conversational AI dialogue systems for Arabic. Our review adheres to the systematic review guidelines outlined by Kitchenham and Xiao in [29]-[30]. Subsequently, we introduce the research questions (RQs) formulated for our systematic review of the problem mentioned above.

RQ1: What is the history of the evolution of the Arabic conversational system and what are the objectives of building conversational chatbots? This question is answered in Section 3. The primary objective of this research question is to illustrate the evolutionary progression of Generative Artificial Intelligence, delving into the development of chatbots and the diverse applications that researchers have explored, particularly in crucial areas, like healthcare or educational support.

RQ2: What approaches are used to perform Generative Arabic conversational AI? This question is answered in Section 7. The aim is to explore state-of-the-art deep-learning techniques.

RQ3: What are the evaluation criteria of the deep-learning techniques used in Arabic conversational AI systems? This question is answered in Section 8. The role of this research question is to specify the measures used to evaluate the deep-learning techniques in the Arabic QA systems.

RQ4: What are the major challenges in building Arabic conversational AI? This question is answered in Section 4 and Section 9. This research question encourages future researchers to explore language-specific techniques by highlighting the significant challenges associated with Arabic conversational AI. The papers reviewed in the various sections were gathered by querying multiple databases, including journal articles and conference proceedings published between 2000 and 2023. The literature collection used the widest publishers, such as IEEE, ACM, Springer, Elsevier, Wiley, Taylor & Francis. Moreover, we searched well-known databases, such as Scopus, Web of Science, DBLP and Google Scholar. Figure 1 shows the percentage of resulting papers per database and Figure 2 illustrates the search results in Scopus from 2000 to 2022 for the keywords TITLE-ABS-KEY ("chatbot*" OR "Generative Artificial Intelligence*" OR "question answering*" OR "question answering system*" AND "Arabic"). The search terms for this review will use various combinations

of search terms derived from the research questions. We derive the principal terms from the research questions about Arabic question answering, Arabic Chatbot and deep learning. The search terms consist of the advanced search string construction using identified keywords' search terms using Boolean operators AND/OR:

1) ("BERT" OR "GAN" OR "GPT" OR "Transformers" OR "Seq2Seq" OR "LSTM" OR "Transformers") AND ("question answering" OR "chatbot" OR "conversational agent" OR "Dialogue System") AND "Arabic".
2) ("Generative Adversarial Networks" OR "Generative Neural Networks" "deep Bidirectional Transformers" OR "DBLSTM" OR "RNN" OR "CNN" OR "DBN" OR "DNN" OR "DANN") AND ("question answering" OR "chatbot" OR "conversational agent" OR "Dialogue System") AND "Arabic".
3) ("deep learning" OR "deep structured learning" OR "hierarchical learning") AND ("chatbot" OR "conversational agent" OR "Dialogue System") AND "Arabic".
4) ("GPT-3" OR "GPT-4" OR "LLMs" OR "LaMDA") AND ("question answering" OR "BARD" OR "conversational agent" OR "ChatGPT") AND "Arabic".

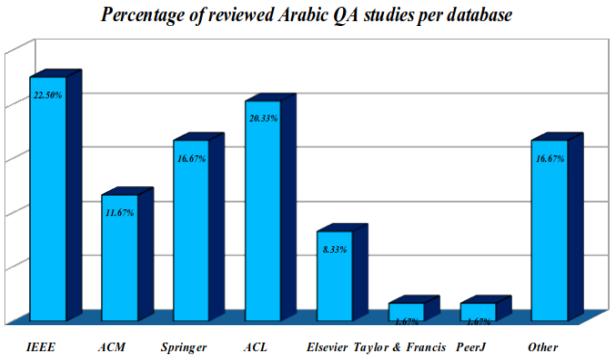

Figure 1. Resulting papers' percentage per database.

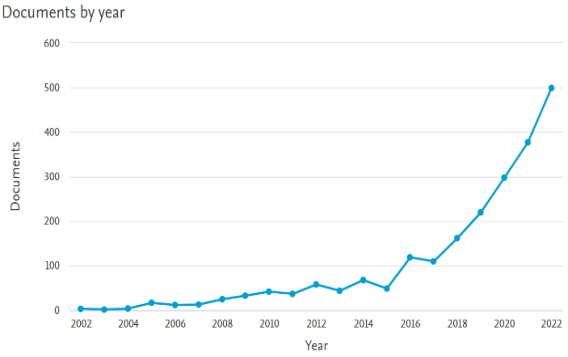

Figure 2. Search results in Scopus, from 2000 to 2022, for the keywords TITLE-ABS-KEY ("chatbot*" OR "Generative Artificial Intelligence*" OR "question answering*" OR "question answering system*" AND "Arabic").

3. CHATBOTS: HISTORY AND CLASSIFICATION

This section presents a comprehensive overview of chatbots, starting with a brief history of their evolution and highlighting the key milestones that have shaped their development. As we delve into this historical chronology, the transformative power of AI-driven advancements becomes evident. Next, we move on to a detailed classification of chatbots, examining various factors that categorize these intelligent entities. Finally, we showcase the use of chatbots in various sectors, highlighting their application areas.

3.1 Chatbots' History

In 1950, Alan Turing proposed the Turing Test ("Can machines think?") [31] to assess a machine's capacity to exhibit intelligent machine behavior similar to humans. To pass the Turing Test, the machine's responses must be indistinguishable from those of a human during a five-minute test. The origin of chatbots dates back to 1966 with the invention of ELIZA [32]. Eliza is a conversational agent in a very basic Rogerian psychotherapist. It was based on a template-based response mechanism and simple keyword matching. Many chatbots were developed after ELIZA, such as PARRY, an infamous chatbot, created in 1972 by Kenneth Mark Colby, a psychiatrist and computer scientist associated with Stanford's Psychiatry Department. JABBERWACKY in 1988 used contextual pattern-matching technique [33]. In 1990, the Loebner Prize was begun [34] (annual competition for chatbots based on Turing Test). In 1992, the authors of [35] developed Dr. Sbaitso's voice-based chabot.

Later in 1995, Richard Wallace [36] developed ALICE (Artificial Linguistic Internet Computer Entity). This chatbot has become significant, because it led to the development of Artificial Intelligence Markup Language (AIML) [37]. AIML is used to declare pattern-matching rules that link user-submitted words and phrases with topic categories. It is an eXtensible Markup Language (XML)-based language and

supports most chatbot platforms and services today. ALICE won the Loebner prize in 2000, 2001 and 2004 [38]. Afterwards, many chatbots were developed based on the ALICE framework [39]. In 2001, SmarterChild chatbot [40] was developed to be compatible with instant messaging applications, such as MSN Messenger and American Online Services (AOI), Instant Messenger (IM) or American Instant Messenger (AIM). In 2005, based on rules written in AIML, Mitsuku (Kuki) [41] was the most widely used stand-alone human-like chatbot. Essential features of Mitsuku are general conversations, which can hold lengthy conversations and multilingual robots that can think logically about a given object. In the Mitsuku chatbot, human curators evaluate incoming data; only the validated data is recorded and used.

Since 2006, new virtual personal assistants have been developed, such as IBM Watson [42], (a rule-based AI chatbot that uses NLP and hierarchical ML methods to generate responses based on the score). Later, many chatbots have been developed, such as Apple Siri [43] (a speech-to-text bot dedicated to Apple products) in 2010, Google Assistant [44] in 2012, Amazon Alexa [19] in 2015, Dialogflow [45] developed by Google in 2016, LUIS [46] developed by Microsoft in 2017 and Amazon Lex [47] developed by Amazon in 2017.

Afterwards, specifically with the introduction of transformers' architecture in 2017 by Vaswani et al. [48], many language model based-transformers were developed, such as BERT (Bidirectional Encoder Representations from Transformers), offered by researchers at Google AI Language in 2018 [27] and GPT (Generative Pre-trained Transformer) developed by OpenAI in 2018 [23]. As a result, generative Artificial Intelligence (AI) has made remarkable advancements in revolutionizing our lifestyle, work dynamics and interactions with technology. Recently, one area that has seen significant progress is the development of Large Language Models (LLMs), such as GPT-3 [25], ChatGPT, GPT-4 [26] and LaMDA, a Language Model for Dialogue Applications [28]. Moreover, these models exhibit remarkable accuracy in tasks like text summarization and question-answering. In Section 7, we study this revelational technique in more detail. The evolution of the conversational agent is shown in Figure 3, along with the evolution of relevant techniques and approaches.

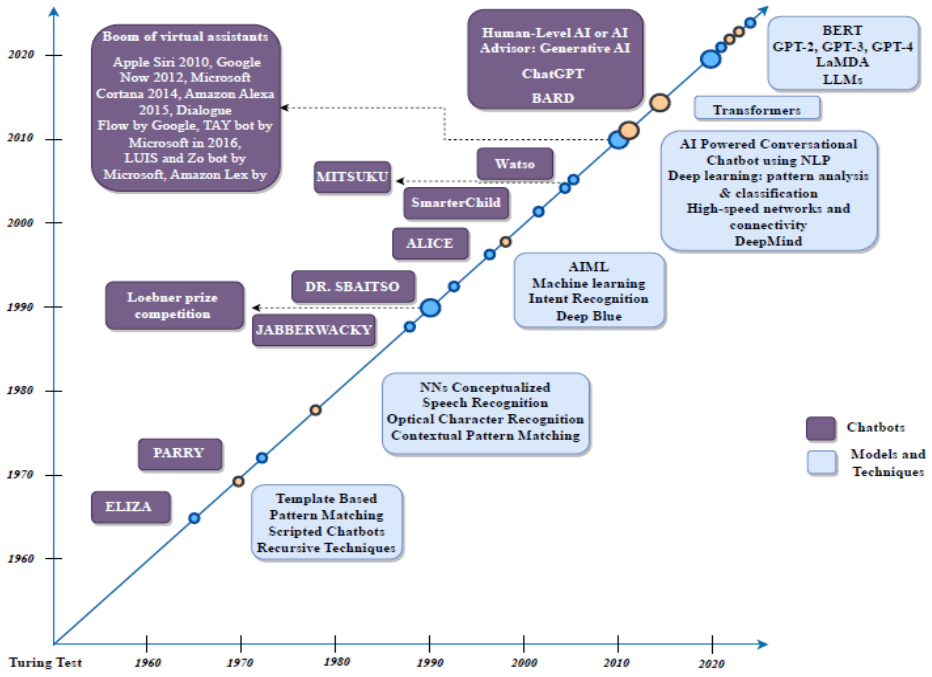

Figure 3. The evolution of chatbots: From ELIZA to generative AI chatbots based on AI advancements.

3.2 Chatbots' Classification

In the last few years, the chatbot field has become so dynamic with the emergence of new technologies that more intelligent systems have developed using complex knowledge-based models. Hence, chatbot classification is essential for scientists to compare and evaluate systems, define requirements and select

the right tools. Figure 4 illustrates our comprehensive broad classification of chatbots, which is based on the main classification proposed by [33], [49]-[51].

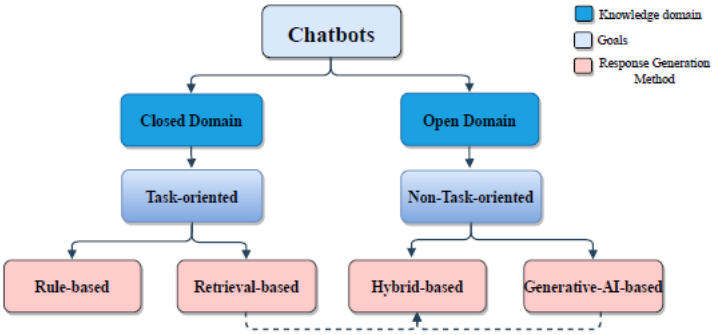

Figure 4. Broad classification of chatbots.

The first superficial level for chatbot classification is based on the following: **Types of questions** [(factoid (when/who/where), confirmation (yes/no), definition, causal (how/why/what), procedural, comparative, opinionated)], **types of knowledge sources** [(structured (RDF graphs, SQL database, CSV, JSON, XML data . . .), unstructured (plain text, e.g. Wikipedia)) and based on **input of user and output of chatbot**, chatbots can be categorized as shown in Table 1.

Table 1. Types of chatbots based user's input and chatbot output.

	Text	**Speech**
Text	Text-to-text bot (TTT): Example: *ELIZA* [32]	Text-to-speech bot (TTS): Example: *Alexa* [52]-[53]
Speech	Speech-to-text bot (STT): Example: *SIRI* [53]-[54]	Speech-to-speech bot (STS): Example: *Cortana* [53]

In Table 2, we see the four main kinds of chatbot classification: the knowledge domain, the response generation method, the goals and the service provided.

Table 2. Taxonomy of chatbot application.

Knowledge domain: Includes the knowledge that a chatbot can access or the amount of data that it is trained upon [49], [55].	Open-domain:A chatbot aiming to establish long-term connections with users who can talk about general topics and respond appropriately. Closed domain:A chatbot operates through information regarding a particular area of interest and aims to provide specific answers concerning only the particular knowledge domain. Generic:A chatbot can answer any user question from whichever domain.
Response-generation method: The distinction is based on the input-processing and response-generation method (the algorithms and the techniques adopted) [56].	Rule-based chatbots: Use a knowledge base organized with conversational patterns, including a list of hand-written responses that correspond to the user's inputs [56]-[57] Three of the most common languages for the implementation of chatbots with the pattern-matching approach are AIML, Rivescript and Chatscript. For these reasons, this model is not robust to spelling and grammatical mistakes in user input. Retrieval-based chatbots: Learn to select responses from the current conversation from a repository with response selection algorithms, heuristics, fairly simple concepts of a rule-based expression match or using a combination of machine-learning classifiers [58]. Also, these chatbots do not produce new responses, but choose one from a pool of predefined responses. Generative chatbots: Generative models are the smartest among the three models in terms of generating answers and can generate more proper responses that could have never appeared in the corpus. These models use machine-learning algorithms and deep-learning techniques. This means that generative chatbots need training with a very large set of data to achieve a good conversation.

Goals: Based on the objectives and the primary goal that the bot aims to achieve, existing dialogue systems are generally divided.	Task-oriented chatbots: are aimed to assist the user with short conversations to complete a particular task, typically used in a closed domain. Non-task oriented chatbots: can simulate a conversation with humans to provide reasonable responses and entertainment. The two major approaches used in non-task-oriented systems are generative and retrieval-based methods. Typically, they focus on conversing with humans on open domains [59].
Service provided: based on the task the chatbot is performing and the amount of intimate interaction that takes place.	Interpersonal chatbots: are usually based on rule-based/ retrieval-based chatbots that offer services like booking servces in restaurants or airlines. Intrapersonal chatbots: exist within the personal domain of the user and understand his/her needs. Inter-agent chatbots: such as Alexa-Cortana integration chatbots to communicate with each other [55].

3.3 Chatbots' Applications

Arabic chatbots have applications in various domains, such as education and healthcare. In the education sector, chatbots can assist students with homework, offer feedback on their work and answer their questions. Examples of these chatbots include the rule-based chatbots [4]-[8] and generative conversational AI [9]-[10]. Moreover, in Section Two, Chapter Four of [11], a review is provided on the various opportunities and challenges associated with educational chatbots. The authors emphasized the advantages of using chatbots in the educational sector, such as their accessibility (24x7 remote access), promotion of self-learning and self-regulation and facilitation of social learning, particularly in creating awareness about societal issues. However, the authors also highlighted specific challenges in implementation. These include issues associated with reliability and accuracy during wide-scale chatbot integration in the learning process, technology limitations in chatbots and insufficient research on various aspects of chatbot technologies. In healthcare, conversational systems like the retrieval-based OlloBot chatbot [12] and AI-based MidoBot chatbot [13]-[15] assist patients in managing their health and answering their medical queries. The potential of conversational dialogue systems to revolutionize human-computer interactions is substantial.

In another study by Abu-Shawar and Atwell [60], a chatbot system called FAQchat is presented. It serves as an interface for Frequently-Asked Questions (FAQ) websites, converting website text into a chatbot-friendly format. The system provides answers using pattern-matching template rules without requiring sophisticated language processing or inference. User trials reveal favorable feedback, with around two-thirds of users preferring FAQ chat over traditional search engines. This demonstrates the practical usability of the chatbot and suggests its potential as a viable alternative for accessing FAQ databases, indicating broader adoption of chatbots in information portal websites.

Artstein et al. [61]-[62] and Traum et al. [63]-[64] introduced New Dimensions in Testimony (NDT). This chatbot application allows users to converse with Holocaust survivor Pinchas Gutter. Developed by the University of Southern California's Institute for Creative Technologies in collaboration with the USC Shoah Foundation, NDT showcases a novel use of AI and natural-language understanding, creating immersive educational experiences. The system goes beyond traditional chatbots by utilizing advanced natural-language processing to generate lifelike virtual avatars of survivors. Based on extensive video interviews, these avatars provide authentic and emotionally impactful storytelling experiences during real-time interactions. While demonstrating the system's effectiveness in simulating conversations with "live" individuals, it has been noted that NDT cannot initiate topics or ask questions in counseling and historical-talk contexts.

In the same context, Abu Ali et al. [65] developed a bilingual (Arabic-English) interactive human avatar dialogue system called TOIA (Time-Offset Interaction Application). The system is inspired by the "new dimensions in testimony demonstration" project by Artstein et al. [61] and simulates face-to-face conversations between humans using digital human avatars recorded in the past. TOIA is a conversational agent based on an actual human being and can be used to preserve and tell stories. The system allows anyone to create an avatar of themselves using a laptop, which facilitates cross-cultural and cross-generational sharing of narratives to wider audiences. TOIA supports monolingual and cross-

lingual dialogues in Arabic and English, but can be extended to other languages. This system has the potential to bridge the gap in dialectal-speech recognition and overcome the challenges of limited resources, lack of standard orthographic rules and lack of definition in Arabic dialects [61][65].

4. CHALLENGES IN ARABIC CONVERSATIONAL AI

This section explores the challenges faced in Arabic conversational AI systems. Based on the literature, we outline general problems related to conversational AI (question-answering systems, chatbots, conversational agents and dialogue systems), like context sensitivity, ambiguity and the need for high-quality labeled datasets. Additionally, we explore Arabic-specific challenges, such as the complexity of Arabic morphology, dialectal variations and the phenomena of Arabizi and transliteration. To visually represent these challenges, we present an illustrative overview in Figure 5.

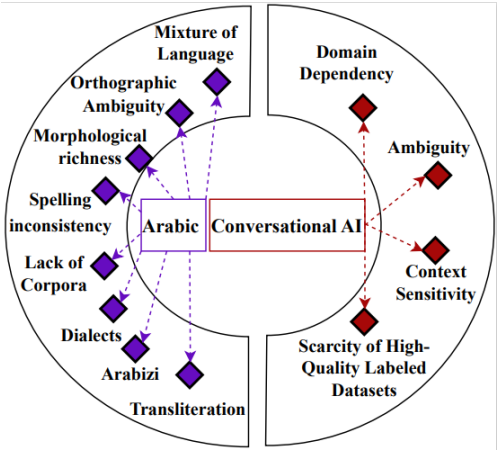

Figure 5. Arabic conversational AI challenges.

4.1 Basic Concepts and General Challenges

Building conversational AI for the Arabic language presents specific challenges due to the complexity and unique characteristics of the language. In this section, we will examine the general problems, which include basic concepts of the Arabic language, ambiguity, data availability and quality, formality and politeness, speech recognition and synthesis and cultural sensitivity.

Basic Concepts: Arabic natural-language processing (NLP) is an increasingly growing field, but it comes with distinctive challenges compared to other languages. Arabic is one of the most widely spoken languages globally, boasting over 421 million speakers as of 2017[1]. Additionally, Arabic encompasses diverse spoken forms [66]-[67]. In his book "On Introduction to Arabic NLP" [68], Habash identifies several challenges that Arabic presents to NLP, including orthographic ambiguity, morphological complexity, dialectal variations and orthographic noise. While some of these challenges may not be exclusive to Arabic, their combination renders Arabic processing notably intricate. Arabic distinctly differs from languages like English in various aspects. For instance, the Arabic alphabet is read and written from right to left and consists of 28 primary characters, 13 of which contain dots, while 15 do not. Furthermore, another specificity differs Arabic from other languages in some punctuation marks, such as the question mark (in Englais '?', in Arabic '؟'), comma (in Englais ',', in Arabic '،') and semicolon (in Englais ';', in Arabic '؛').

Cultural Sensitivity: Arabic conversational AI systems need to be culturally sensitive and avoid generating responses that may be considered offensive or inappropriate in the Arab world. Understanding cultural norms and values is crucial for building successful conversational AI systems. However, the development and training of Arabic conversational AI systems might suffer from a lack of multicultural representation. If the AI system is primarily designed and trained by individuals from a limited cultural background, this could result in a system biased towards that particular culture and less inclusive of the diverse perspectives and values [69].

Speech Synthesis and Recognition: Compared to English, Arabic voice-based conversational AI

[1] Source: https://www.internetworldstats.com/stats19.htm

systems have less advanced Text-to-Speech (TTS) and Automatic Speech Recognition (ASR) technology. According to [70], the limited availability of Arabic ASR and TTS systems is evident compared to other languages, indicating a relative scarcity of high-quality systems. This lack of resources hinders the development of reliable speech-recognition and synthesis capabilities for Arabic Conversational AI. Moreover, Arabic poses unique challenges in phonetics and pronunciation due to its complex phonetic system, encompassing a wide range of sounds and phonetic variations [68][71]. This complexity creates difficulties for ASR systems to accurately recognize and transcribe Arabic speech, particularly when handling various accents, dialects, limited vocabulary and diacritic marks. These diacritic marks are vital in providing contextual information, making it challenging for ASR systems to achieve precise transcriptions without them.

Ambiguity, Homonymy and Gender Agreement: These aspects pose challenges in the context of Arabic AI systems, particularly in speech recognition [72] and chatbot applications. Arabic nouns, pronouns and verbs exhibit gender agreement, meaning that they vary based on the gender of the speaker and the entities being referred to. This presents difficulties in developing gender-inclusive and culturally sensitive AI systems that respond appropriately to diverse users. Additionally, Arabic words often have multiple meanings depending on the context [70], [73]-[74], leading to ambiguity challenges. Homonymy, where different words share the same pronunciation but have distinct meanings (Table 3), further complicates the accurate interpretation of user input and the provision of appropriate responses.

4.2 Arabic-specific Challenges

In addition to the broader challenges facing conversational AI, there are also specific difficulties related to the Arabic language's various dialects and morphological structure. Since conversational systems heavily rely on the morphology of the target language, as noted by studies such as [11] and [75], it is essential to consider the unique linguistic characteristics of Arabic, including its dialects, orthography and morphology.

Arabic Varieties: Figure 6 illustrates the three main varieties of Arabic [66]-[68]. The first is classical Arabic, known as Quranic Arabic, used in religious texts and various old Arabic manuscripts. The second variety is Modern Standard Arabic (MSA), the formal means of communication that most Arabic speakers understand. MSA is commonly employed in newspapers as well as in radio television broadcasts. The third type is dialectical or colloquial Arabic, which is utilized in everyday conversations and has recently found its way into TV and radio broadcasts. Arabic dialects can be categorized into five main groups based on geographical distribution (Maghrebi dialect, Egyptian dialect, Levantine dialect, Iraqi dialect and Gulf dialect). However, it is essential to note that this classification is general and relies on the proximity of countries, leading to commonalities in dialectical words and expressions.

Additionally, Farghaly et al. [66] and Ryding et al. [67] proposed a classification of dialectal Arabic into two main categories: western and eastern. The reasons for the existence and growth of many Arabic dialects are geographical, social, political and primarily linguistic conflicts. The advantage of dialect lies in the economy of language, characterized by abandoning common expressions and forms, ignoring, quoting and updating meaning. The nature of life is willing to ignore what should be ignored to quote what is necessary.

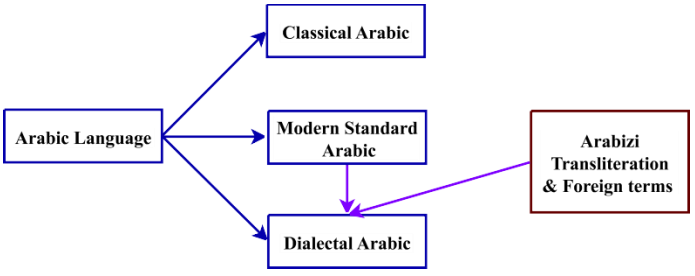

Figure 6. Arabic language varieties.

Arabic Ambiguity and Orthography: The Arabic orthography allows optional diacritical marks to denote short vowels and consonantal doubling. These diacritical marks provide additional information

about the pronunciation and structure of words in the written text. However, people commonly omit these marks from the text and do not use upper and lower cases, leading to high ambiguity. Additionally, according to [76], Arabic writers often need to correct spelling, particularly with problematic letters like Alif-Hamza "أ آ" forms and Ta-Marbuta "ة". The problem of orthography is even more challenging for Arabic dialects, as they need standardized orthographies [74][77]. Diacritical marks are used in Arabic to aid pronunciation and clarify the meaning of words. However, proficient speakers can understand the text without these marks, so most Modern Standard Arabic (MSA) texts are written without them, resulting in lexical ambiguity. Diacritical marks include point diacritics, short vowels (fatHah, kasrah and DHammah), linguistic diacritics and decorative marks. This diversity can lead to the problem of having several diacritics on a single base letter. Diacritics are crucial in distinguishing between orthographically similar words, but disambiguated by diacritics. Table 3 illustrates three examples of this.

Table 3. Orthographically similar words disambiguated by diacritical marks.

Triliteral roots	Meaning-1	Meaning-2	Meaning-3
ع، ل، م	عَلِمَ 'to know'	عَلَم 'flag'	عِلْمْ 'science'
ح، س، ب	حَسَبَ 'to calculate'	حَسِبَ 'to think / assume'	حَسَب 'decent', 'according to'
ك، ت، ب	كَتَبَ 'to write'	كُتُب 'books'	كَتَّبَ 'to force to write'

Morphological Complexity: Arabic exhibits many inflectional characteristics, in which words undergo various forms based on their context within a sentence, including inflections and derivations. Developing a robust morphological analyzer, a vital component of any natural-language processing system [78] becomes challenging due to this linguistic feature. Habash [68] and Habash-Soudi-Buckwalter [78] highlighted that morphological analysis stands out as one of the most demanding tasks in Arabic NLP due to the language's rich morphological structure encompassing a considerable number of allomorphs and exceptions.

To address these challenges, researchers are developing new techniques and technologies to enhance the accuracy and effectiveness of Arabic conversational AI systems. For example, Habash-Soudi-Buckwalter [79] introduced an Arabic transliteration (HSBT) scheme. Arabic transliteration involves representing Arabic text using Latin characters, providing a phonetic representation of the script. This enables non-Arabic speakers or those unfamiliar with the Arabic script to read and pronounce Arabic words. Transliteration finds applications in language-learning materials, dictionaries and communication between Arabic and non-Arabic speakers. The HSBT transliteration scheme [79] extends Buckwalter's original scheme [80], which designates a single, distinct ASCII character for each Arabic letter. HSBT enhances readability by incorporating some non-ASCII characters while maintaining a distinct 1-to-1 mapping between Arabic and Latin characters. For example, the Arabic word السَلَام عَلَيكُم, when transliterated with the HSBT scheme, is written as "As-salāmu alaykum".

An Arabic word manifests various morphological aspects, such as derivation and inflection. Multiple words with distinct meanings originate from the same root in derivation. For instance, from the root "قَلب" ('heart'), we derive "قُلُوب" ('hearts') and "قَالِب" ('Mold/Template'). On the other hand, the inflection aspect involves varying the same word to denote different grammatical categories expressing the same meaning. For example, the root word "قَرَأَ", meaning 'to read' in Arabic, appears as "أَقرَأُ" in the present tense, first person singular form and as "قَرَأتُ" in the past tense form.

Arabic's intricate morphology and unique language features have introduced additional challenges for building Conversational AI, particularly the need for more Arabic question-answering resources. We will now delve into the obstacles linked to the complexity and diversity of the Arabic language, including the lack of corpora, the presence of dialects and Arabizi and the mixture of languages.

Limited Resources and Data Availability: Building a robust and accurate conversational AI system necessitates vast amounts of annotated data for training and evaluating NLP models. The effectiveness of conversational AI is directly linked to the quality and size of the corpus used for training. According to [81]-[82], the availability of Arabic question-answering datasets is limited. Moreover, a comparative analysis of English and Arabic languages in building text-based conversation agents was explored in [73]. The study revealed that constructing conversational AI systems in Arabic is notably more challenging due to the language's complexity and the scarcity of available resources. Additionally, the authors of [73] pointed out

that Arabic speakers often use different linguistic forms based on the conversation's context, sometimes even within the same conversation. This phenomenon, known as diglossia, arises when multiple forms of the same language coexist within the same speech community.

Furthermore, the paper by Antoun et al. [83] highlighted that numerous datasets are derived from translations of other languages, often relying on resources like English-SQuAD, TREC or CLEF. The authors observed that these translated datasets, such as Arabic-SQuAD and ARCD, include text elements in languages other than Arabic, encompassing unknown sub-words and characters. This issue arises from inadequate training samples translated from the English SQuAD dataset. Moreover, Alwaneen et al. [81] pointed out that utilizing translated datasets can introduce language-related complexities. Most Arabic QA datasets are limited, making it impractical to directly compare different systems, as each uses its unique dataset. Furthermore, authors are generally reluctant to share their working code. However, a few exceptions exist, such as the Arabic-squad [82] and TyDi-QA [84] datasets, which incorporate the Arabic language.

Dialectal Variation: Arabic is spoken in many different dialects, each with its distinct vocabulary, grammar and pronunciation. Consequently, constructing a single conversational AI system capable of effectively communicating with users from diverse Arabic-speaking regions presents a considerable challenge [73], [85]. The variations in pronunciation, vocabulary and grammar across these dialects pose significant limitations for NLP systems attempting to process and comprehend all forms of Arabic. Furthermore, Habash's book [68] emphasizes the complexities of dialectal variation in natural-language processing tasks, underscoring the difficulties faced in handling such diversity.

The Arabizi Phenomenon: Arabizi is an informal and non-standardized form of writing that involves using the Latin alphabetic transliteration or writing of Arabic words [68], [86]. It is a term used to describe the practice of phonetically writing Arabic words using a combination of Latin characters and numerals. Arabizi is commonly used in informal digital communication, including social media, chat applications [75] and text messages. This writing style appeals to young Arabic speakers and those who feel more at ease using Latin characters while communicating in Arabic. For example, the number "3" represents the letter "ع" in transliteration, while the number "7" corresponds to the letter "ح". In Arabizi, the Arabic phrase اَلسَلَام عَلَيكُم is written as "as-slm 3lykm".

No Capital Letters and Code Switching in Different Alphabets: Arabic lacks uppercase or lowercase letters, which poses a challenge for factoid question systems that rely on named entity recognition to identify proper names, places and person names [87]. This limitation makes it difficult to differentiate between proper Arabic nouns and other word forms, like adjectives and common nouns [88]. Additionally, Arabic users frequently mix dialectal expressions and Arabizi within their discussions. As both dialects and Arabizi lack standardized rules, they can be written differently, presenting similar linguistic challenges for conversational AI systems.

Furthermore, Arabic speakers are often bilingual or multilingual, particularly in science and medicine. Consequently, users may ask questions that contain more than one language. Pre-processing questions or answers by filtering out Latin letters could result in a loss of meaning, leading the question-answering system to provide inaccurate or empty responses.

4.3 Arabic NLP Tools and Resources

The complexities and characteristics of the Arabic language present a significant challenge for researchers and developers in processing Arabic text. Dealing with ambiguity, diglossia and comprehending the Arabic script necessitates using specialized Arabic NLP tools and resources. These dedicated tools are designed to tackle the unique challenges of Arabic-language processing, ultimately enhancing the accuracy and efficiency of NLP tasks. This sub-section introduces some widely utilized Arabic NLP tools, including MADAMIRA, Farasa and CAMeL, along with the notable online Masader+ catalog of Arabic NLP data and the Special Interest Group on Arabic NLP (SIGARAB). These valuable resources collectively aid in addressing the challenges above and contribute to advancing Arabic-language processing.

MADAMIRA is an Arabic morphological text-analysis tool developed by researchers [89], that combines the two widely-used systems in Arabic pre-processing, MADA and AMIRA [78], [90]-[91], to provide a comprehensive tool for pre-processing Arabic text. This fusion has created the powerful MADAMIRA tool, specifically designed to cater to various linguistic analysis tasks essential for Natural

Language Processing (NLP) in Arabic. Among the key features which MADAMIRA provides are tokenization, lemmatization, part-of-speech tagging, Base Phrase Chunking (BPC) and Named Entity Recognition (NER). Its tokenization capability ensures that input Arabic text is effectively segmented into individual units, preparing it for further analysis. With part-of-speech tagging, MADAMIRA assigns grammatical categories to each token, facilitating a deeper understanding of the sentence's syntactic structure. Additionally, MADAMIRA provides BPC for identifying phrases and their syntactic roles within the sentence and NER capabilities, enabling identifying and extracting named entities, such as names of people, locations, organizations and other relevant entities in the text.

Farasa, an Arabic segmenter developed by Abdelali et al. [92] uses SVM-rank to learn feature vectors for each segmentation. The segmentations are then scored with the trained classifier. The tool comes in two varieties: FarasaBase and FarasaLookup. FarasaBase utilizes a classifier to segment words with a lookup list containing concatenated stop-words. On the other hand, FarasaLookup involves a training process where seen segmentations are cached during training and classification is applied to unseen words. Farasa supports several key features and functionalities, including Dialectal Analysis, Diacritization, Text Normalization, POS tagging and Named Entity Recognition (NER). Furthermore, Farasa has demonstrated its performance in Machine Translation (MT), achieving an accuracy rate of 98.94%, outperforming MADAMIRA in this task.

Recently, Obeid et al. [93] introduced **CAMeL** Tools, a comprehensive set of tools and libraries designed explicitly for Arabic natural-language processing tasks. This toolset supports Arabic and Arabic dialect pre-processing, providing many features, such as transliteration orthographic normalization, discretization, dialect identification, morphological modeling, sentiment analysis and named entity recognition. The development of CAMeL Tools was motivated by addressing limitations in previous rule-based systems, such as fragmented tasks spread across different systems and a need for more flexibility. By integrating diverse functionalities into a well-suited toolkit, CAMeL Tools presents a more efficient and flexible approach for tackling Arabic NLP tasks. Additionally, the toolkit includes libraries that facilitate working with diverse text formats like HTML and XML and text classification and clustering capabilities. Experiments conducted on CAMeL Tools demonstrated significant performance improvements compared to some state-of-the-art models and it outperforms MADAMIRA in various processing utilities. Furthermore, Antoun et al. antoun2020arabert utilized CAMeL Tools to fine-tune the pre-trained language model AraBERT, showing superior results compared to the CRF-based system in the named entity recognition task.

Recently, the importance of a corpus as a fundamental component for building accurate question-Answering systems has become increasingly evident. Consequently, researchers have made significant efforts to develop more accessible Arabic language resources. One notable platform in this regard is Masader+, an online catalog of Arabic NLP data, which serves as a comprehensive repository of linguistic resources and datasets for the Arabic language [GitHub[2]]. Developed in 2022 by Alyafeai et al. [94], Masader+ represents an updated version of the original Masader catalog. It offers a wide range of NLP data, including corpora for named entity recognition, question-answering systems and sentiment analysis tasks. The data in the catalog is freely available for researchers and developers working in Arabic NLP. The website provides detailed information about each dataset in the catalog, including data size, format, suitability for specific tasks and data source. Additionally, it offers instructions for data downloading and usage, along with links to related resources and publications. Masader+'s organized and user-friendly interface ensures easy access to high-quality datasets, fostering research, development and evaluation of NLP models and applications tailored to Arabic. As a valuable resource for the latest developments in Arabic NLP, the Masader+ website proves to be beneficial for anyone interested in advancing language processing in the Arabic domain.

The MADAMIRA, Farasa and CAMeL Tools, alongside the online Masader+ catalog of Arabic NLP data, demonstrate remarkable versatility, efficiency and improved performance, making them invaluable resources for researchers and developers in Arabic NLP. With their diverse features and continuous enhancements, these tools play a significant role in advancing Arabic-language processing and enabling the creation of sophisticated and accurate NLP applications for Arabic. Notably, all three tools are open-source and offer command-line interfaces (CLIs) and application programming interfaces (APIs), offering users convenient and flexible integration into their NLP workflows. Additionally, we have the Special Interest

[2] Website available from https://arbml.github.io/masader/

Group on Arabic NLP (SIGARAB), a dedicated professional organization operating under the Association for Computational Linguistics (ACL). SIGARAB strives to promote the growth of Arabic NLP technologies, fostering knowledge exchange among researchers and practitioners. The organization showcases state-of-the-art research in Arabic NLP through regular meetings, conferences, newsletters and proceedings. Its website[3] serves as a valuable resource, providing information on news, events and diverse resources, such as datasets, software and documentation for researchers and developers in the Arabic NLP field.

5. RULE-BASED AND RETRIEVAL-BASED CHATBOTS

In contrast to English and other chatbots, Arabic still needs to be more powerful and efficient. Moreover, Arabic chatbots are comparably scarce due to the challenges coming from the language itself. The challenges originate from the uniqueness of the Arabic representation style, the richness of its morphology, the different meanings of each word and the synonyms provided to express a specific request.

The authors in [4] proposed ArabChat, a rule-based conversational agent developed using the PM approach. In the same context, the authors in [5] proposed a mobile version of ArabChat and the authors of ArabChat [4] provide the "Enhanced ArabChat" [6]. Authors of [8] have simulated Nabiha, a rule-based chatbot based on PM and AIML approaches. Ollobot is an Arabic rule-based chatbot proposed by Fadhil et al. [12] using the AIML method. In [75], the writers stated that they developed a retrieval-based chatbot called BOTTA using several AIML files. Aljameel et al. [7] proposed LANA as an Arabic retrieval-based chatbot based on a combination of Pattern Matching (PM) and Short Text Similarity (STS) to extract the responses. In 2021, an Arabic flight booking dialogue system was proposed using rule-based and data-driven hybrid approaches in [21]. More recently, in 2022, AlHumoud et al. presented Rahhal [20], an Arabic rule-based chatbot for helping tourists visit different Saudi Arabian cities based on the PM approach. The approaches applied in previous works are based essentially on pattern matching. However, these techniques are based on something other than grammatical or linguistic details. The rule-based system is the oldest in chatbot development. Recent advances in Artificial Intelligence (AI) and Natural Language Processing (NLP) enable data-driven approaches to developing chatbots. Table 4 summarizes the pre-mentioned Arabic Chatbot systems.

Table 4. Rule-based and retrieval-based Arabic dialogue system/chatbots.

Chatbot	Application domain	Implementation Technique	Response-generation technique
ArabChat [4]	Closed Domain (assisting the students of ASU)	Pattern Matching	Rule-based
Mobile ArabChat [5]	Closed Domain (assisting the students of ASU)	Pattern Matching	Rule-based
Enhanced ArabChat [6]	Closed Domain (assisting the students of ASU)	Pattern Matching	Rule-based
Botta [75]	Open Domain	AIML	Retrieval-based
Nabiha [8]	Closed Domain (assisting the students of KSU)	AIML and Pattern Matching	Rule-based
LANA [7]	Closed Domain (science for children with autism)	STS and Pattern Matching	Retrieval-based
OlloBot [12]	Closed Domain (Healthcare)	AIML	Retrieval-based
Flight booking [21]	Closed Domain (airline-ticket booking)	Pattern Matching	Rule-based and data-driven
Rahhal [20]	Closed Domain (helping tourists at Saudi Arabia)	Pattern Matching	Rule-based

6. DEEP LEARNING-BASED CHATBOTS

Deep learning is a discipline of machine learning. It is known for learning embedded and abstract representations from raw data with minimal human intervention. Deep-learning models can process large datasets efficiently, saving time by eliminating human intervention and feature engineering [95].

[3] Group available from http://www.sigarab.org/

Deep learning has been recently investigated for the Arabic question-answering system and has shown excellent performance in this area [96]-[97]. These studies use vector-pre-trained word representation datasets for these models, such as Word2vec, Glove and Fasttext, which learn from unlabeled text. In addition, more advanced language representation models, such as BERT [98]-[99] and ELMO [100], have also been used. For Arabic question-answering systems, only a few recent works have explored deep-learning models. Table 5 presents the significant works on deep learning-based Arabic QA systems.

Table 5. Deep learning-based Arabic question-answering systems/chatbots.

Ref.	Year	Domain	Approach	Dataset	Result
[101]	2020	Open Domain	Learning-based (AraBERT)	Arabic-SQuAD and ARCD	AraBERT v1: F1: 82.2% SM: 95.6% EM: 54.8%
[102]	2020	Closed Domain (Medical)	Learning-based (DNN, PCA)	SemEval-2017 CQA-subtask D	MAP: 62.90% MRR: 68.86% AvgRec: 86.6%
[82]	2019	Open Domain	Learning-based (BERT)	ARCD	F1: 27.6% EM: 12.8% SM: 29.8%
[103]	2018	Closed Domain	ML classifier	stack overflow [104]	Random forest Prec: 71.1% Recall: 74.1%
[13]	2022	Closed Domain	DL (seq2seq)	variety of source	N/A
[98]	2021	Closed Domain	BERT, CNN Word2Vec	D1: SemEval-2016 task 3 Subtask B. D2: Quora dataset	BERT/word2vec/CNN+feature D1: Acc:79.57% F1: 71.58% D2: Acc: 88.80% F1 83.18%
[100]	2020	Open Domain	Learning-based (ELMO, CNN, RNN)	Collected manually	Weighted F1: 94% F1: 94%, Acc: 94%
[105]	2021	Open Domain	Learning-based (SVM)	Collected from TREC, CLEF and Moroccan school books	Acc: 90%, Prec: 91% F1: 90%, Recall 90%
[106]	2022	Open Domain	Learning-based (LSTM, CNN, W2V)	Translated from [107]	ASLSTM in Arabic: P@5: 0.37% P@10: 0.28%, MAP: 0.45% Recall: 0.41%
[108]	2020	Open Domain	Semantic and logic-inference-based	AQA-WebCorp [109]	Acc: 0.74%
[110]	2021	Closed Domain	AI-based	-	N/A
[111]	2022	Open Domain	Deep-based DPR, AraELECTRA	491,253 doc: Arabic Wikipedia ARCD [82] TyDiQA GoldP [84]	Single training [84]: EM: 41.8%, F1: 50.1% Single training [82] EM: 15.1%, F1: 35.3% Multi training [84] EM: 43.1%, F1: 51.6% Multi training [82] EM: 15.7%, F1: 36.3%

In fact, Antoun et al. [101] introduced AraBERT, a specialized variant of the BERT (Bidirectional Encoder Representations from Transformers) model designed specifically for Arabic language. The base model consists of 12 attention heads, 12 encoder blocks, 768 hidden layers, a maximum sequence length of 512 and 110M parameters. A large dataset of 70 million sentences, equivalent to 24 GB of text, was used to train the model. Next, the authors conducted evaluations of the pre-trained model on three downstream Arabic tasks: Arabic question-answering, sentiment analysis and named entity recognition. Finally, the authors fine-tuned the pre-trained language model for question answering using two datasets, Arabic-SQuAD and ARCD [82].

Al-Miman et al. [102] proposed a deep neural network model for the answers' ranking problem in Arabic language using the provided Arabic dataset in SemEval-2017 CQA-subtask D. The authors try to find and integrate different types of similar features. The deep model uses the results of the integrated feature set as input. The ranking positions are then generated from the question-answer pairs. The proposed model incorporated features at three levels. The first level utilized Principal Component Analysis (PCA) features, while the second and third levels incorporated similarity features before and after pre-processing, respectively.

The authors of [82] developed a deep-learning system called SOQAL, an open-domain Arabic question-answering system. Their system is based on integrating TF-IDF with a multilingual pre-trained Bidirectional Transformer (mBERT) neural Machine Reading Comprehension (MRC) model to answer open-domain fact queries. TF-IDF is used to group the retrieved documents; i.e., the documents most relevant to the query are selected and responses are extracted from these documents using a multilingual pre-trained Transformer mBERT bidirectional model. A benefit of this research is that the authors first enrich the Arabic research community by creating a corpus that can effectively serve as a training resource for Arabic QA systems. Second, increasing the corpus size can improve the end-to-end QA system. Finally, the authors point out the possibility of improving the system to obtain correct answers using paragraph selection.

Elalfy et al. [103] proposed a hybrid approach using two modules, called content-based and non-content-based modules, to find the best answers on CQA websites. This work used content features, question-answers, answer-answer features and the user-reputation score.

Boussaksso et al. [13] presented an Arabic chatbot called MidoBot, based on the Seq2Seq model, to generate new responses from a dataset. They used AraVec [112] for embedding and the dataset consists of 81,659 lines collected using blogs, plays, Quora Arabic and movie subtitles. Van Tu, N. et al. [98] proposed a model based on the BERT model that integrates various features from other methods.

Hamza, A. et al. [100] proposed a deep model based on distributed word representations (ELMo embeddings) and deep neural (CNN and RNN) models for Arabic question classification (QC). The authors proposed classifying questions into seven categories by finding syntactic and semantic relationships between words and using ELMO to represent the questions. Additionally, they used a dataset of 3,173 Arabic questions, collected and annotated manually, to evaluate their system.

The same authors proposed in [105] a framework based on a machine-learning approach and words' continuously distributed representation for Arabic question classification. First, they proposed a taxonomy of open-domain questions in Arabic, where they represent questions using TF-IDF weighting with $n-$ grams and word representations with a bag of its character $n-$ gram for represented the questions to find the syntactic and semantic relations between words. Then, in the experimentation phase, they used a dataset of 1.302 questions collected from TREC, CLEF and Moroccan school books and they used the SVM algorithm to classify questions.

Othman, N. et al. [106] proposed a deep-learning approach called ASLSTM (Attentive Siamese LSTM-based approach) to tackle similar question-retrieval problems. The ASLSTM method is based on a Siamese architecture with a long short-term memory (LSTM) network, supplemented by an attention mechanism, which enables the model to give extra attention to different words when modeling the questions. To evaluate the proposed approach for the Arabic language, they used Yahoo!Webscope dataset[4], which was translated into Arabic using Google Translate, comprising 1,256,173 questions and 12,512,034 different words. They trained the translation dataset for Arabic word embedding training using Word2Vec (CBOW model); window size = 300 and LSTM training; layer size = 50; embedding layer size = 300.

Bakari, W. et al. [108] proposed a system based on logic form and conceptual graph approach for factoid Arabic question-answering systems. Their system is based on an intuitive understanding of Arabic texts to convert them into semantic and logical representations. First, they used conceptual graphs to transform Arabic text to obtain a logical representation and then, they extracted answers based on the relationship between questions and text passages. They evaluated their system through the NArQAS system [113] using the question-text corpus (AQA-WebCorp) [109]. Therefore, their logic-based approach combined with textual participation (RTE) detection significantly improved the performance of Arabic QA systems.

Al-Madi, N.A., et al. [110] developed an intelligent Arabic chatbot system. This social chatbot can assist students from the AlZaytoonah Private University of Jordan and answer their questions regarding their educational progress using the spoken Arabic language.

Alsubhi, K. et al. [111] helped improve the performance of the Arabic open-domain question-answering system. They used deep-learning techniques to build their end-to-end Arabic open-domain question-answering system based on 491,253 documents from Arabic Wikipedia articles. The model consists of two stages; the first is a passage retrieval task, where they retrieved the top 20 passages relevant to the question using ARCD [82] and TyDiQA GoldP [84] datasets using DPR (Dense Paragraph Retrieval) [114]. Then, for the reading comprehension task, they connected DPR to the AraELECTRA [83] passage reader to obtain the first three answers.

AraELECTRA [83] is an Arabic language representation model pre-trained using the Replaced Token Detection (RTD) methodology on a sizeable Arabic text corpus. Dense Passage Retrieval (DPR) [114] is an efficient retrieval method for open-domain question-answering tasks that uses dense representations to compute relevancy. These methods use deep neural networks to embed documents and questions into a shared embedding space. Dense models use transformer-based encoders that are more sensitive to features, such as lexical changes or semantic relationships.

[4] https://https://maktoob.yahoo.com/?p=us/

Due to their complexity, deep learning-based approaches have shown great promise for improving Arabic question-answering systems. Multiple layers in the deep-learning models effectively decompose Arabic text for more accurate understanding and analysis. We find that Arabic language has had a different luck than English language regarding using the deep-learning approach during the development of QA systems [115]. For these reasons, we will give more importance to the consideration of contextualized word embedding, large language models and deep-learning models for building Arabic QA systems. Furthermore, we find that most existing deep learning-based approaches for Arabic QA have focused on Modern Standard Arabic (MSA), with some using datasets derived from translations of other languages, such as English-SQuAD, TREC or CLEF. However, [81][83] noted that Arabic-SQuAD and ARCD datasets may contain text elements in languages other than Arabic, including sub-words and unknown characters. This highlights the importance of developing adequate representations of words for deep-learning models, which can significantly impact their performance. Following the recent works [82][101], where they proposed two pre-trained Arabic language representation models (AraBERT and AraELECTRA) based on the BERT architecture, specifically designed for Arabic language, we believe that these advances demonstrate the potential of deep learning-based approaches to improve QA systems in Arabic and contribute to the growth of this field.

7. DEEP LEARNING TECHNIQUES

Recently, many algorithms based on deep-learning techniques have made significant progress, aiming to solve the related problems of question-answering systems (task-oriented and non-oriented dialogue systems, answer modeling and answers' ranking problems) based on word representation like non-contextual/ contextual Word Embedding and Uni/Bi-directional model by learning feature representations in a high- dimensional distributed fashion and achieving remarkable improvements in these aspects. Figure 7 shows the evolution of deep-learning techniques.

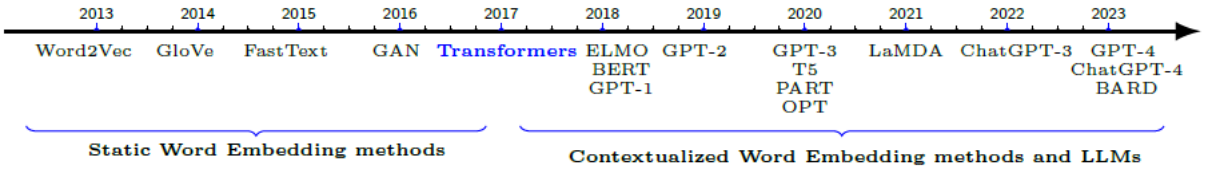

Figure 7. Deep-learning techniques' evolution timeline (2013-2023).

Using distributed representations such, as contextualized or static word embeddings, numeric vectors represent words with similar meanings and contextual usage. Pre-trained embeddings, a successful application of unsupervised learning, can capture semantic and syntactic information, enhancing the performance of various downstream tasks [116]-[117]. A key advantage of pre-trained embeddings is their independence from costly annotation, as they can be derived from extensive unannotated corpora readily available. Consequently, these pre-trained embeddings can be leveraged in downstream tasks that involve limited amounts of labeled data.

7.1 Static Word Embedding

Static Word Embedding methods train the models based on the co-occurrence statistics, such as Word2vec [118], GloVe [119] and FastText [120], which are critical components in many neural language-understanding models.

Word2Vec was introduced by Tomas Mikolov et al. at Google Research [118], helping frame the distributional hypothesis in a predictive approach. Word2Vec has two primary model architectures: the Continuous Bag of Word (CBOW) and the Skip-Gram (SG). These models are algorithmically similar, except that CBOW predicts the target words from the source context words, while Skip-Gram predicts the source context words from the target words. For a set of words in a context window, CBOW sums the vectors representing these words to produce a vector representation of this context. Skip-Gram represents each word and context as d-dimensional vectors to produce similar vector representations of similar words. The CBOW model works better than Skip-Gram on syntactic tasks and much better on the semantic part [121].

Similar to Word2Vec, **GloVe** (Global Vectors for Word Representation) proposed by Jeffrey Pennington [119] is another word vector representation technique. GloVe employs a count-based approach to reduce the dimensionality of the co-occurrence counts' matrix. The main concept behind GloVe is to construct a comprehensive co-occurrence counts' matrix from a given corpus, where each cell in the matrix represents the frequency of a word occurring within a specific context. By normalizing the values for each row of the matrix, we obtain the distribution probability of each context for a given word. Like Word2Vec, word similarity is determined by the similarity between their corresponding vectors. When it comes to dialogue systems, various existing approaches have been utilized. These include retrieval-based methods [122]-[124], as well as generation-based models [125]-[126].

FastText, an open-source project developed by Facebook AI Research Lab, is utilized for constructing scalable solutions in text representation and classification [120]. It serves as an extension to the Word2Vec model and offers an alternative word-embedding approach. Unlike directly learning vectors for individual words, FastText represents each word as an n-gram of characters.

7.2 Contextualized Word Embedding

Unlike traditional word embeddings, such as word2vec or GloVe, contextualized word embeddings consider the surrounding words and sentence structure to generate more nuanced representations. Those approaches have shown effectiveness in various NLP tasks, including sentiment analysis, named entity recognition, machine translation and question-answering. The popular approaches for generating contextualized word embeddings include models like ELMO and transformer-based models like OpenAI's GPT and Google's BERT.

Transformers introduced in 2017 by Vaswani et al. [48] is a type of deep-learning model architecture that has revolutionized various natural-language processing tasks. The transformer has become the unavoidable architecture that forms the basis of the LLMs due to its powerful capabilities. According to [127]-[128], the critical component of a transformer is the self-attention mechanism, also known as scaled dot-product attention. It allows the model to weigh the importance of different words in a sequence when making predictions or generating outputs. In addition, transformers leverage self-attention mechanisms to capture dependencies between words or tokens in parallel. This advantage enables them to effectively model long-range dependencies and capture global context, making them particularly well-suited for tasks involving large text sequences.

LLMs stand of Large Language Models, such as BERT [27] and GPT-3 [25], refer to models trained on a large corpus of text data using unsupervised learning techniques with a massive number of parameters. LLMs are designed to generate coherent and contextually relevant text across various natural-language processing tasks. These models have been widely used for question answering, language translation, text completion and text generation.

LaMDA refers to Language Model for Dialogue Applications [28] as a specialized language model focused on improving dialogue-based applications. LaMDA is explicitly designed for dialogue applications and has the potential to generate responses and improve conversational AI systems and chatbots like Bard. It aims to enhance the flow and coherence of dialogue interactions by addressing challenges, like context understanding, ambiguity resolution and generating more natural and contextually appropriate responses.

ELMo (Embeddings from Language Models) introduced by Matthew Peters et al. [129] is a deep contextualized word-representation model. ELMo word vectors are computed using bidirectional LSTMs (Long Short-Term Memory Networks) to generate word embeddings that capture contextual information.

BERT (Bidirectional Encoder Representations from Transformers), offered by researchers at Google AI Language [27], is a language representation system established with multi-directional language modeling and attention mechanisms. The principal originality of the system is the pre-training approach that captures word and sentence-level representations through Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) training. BERT is pre-trained in diverse languages by employing available unlabeled data. Furthermore, the pre-trained deep bidirectional model has become state-of-the-art in multiple NLP applications like question answering. The concept is to give a general model suitable for diverse applications and a pre-trained architecture that minimizes the necessity of annotated

data. For example, for a provided word, its embedding vector is constructed by counting the embeddings of the related word, the sub-words and the position together.

T5 stands of Text-To-Text Transfer Transformer introduced by Raffel et al. in 2019 [130]. It is a transformer-based model with significantly advanced natural-language processing tasks. The T5 model follows a unified text-to-text framework, where a wide range of language tasks, such as translation, summarization and question-answering, can be formulated as a text-to-text conversion problem.

GPT, short for Generative Pre-trained Transformer, is a family of unsupervised transformer-based generative language models developed by OpenAI. The GPT models comprise several large language models (LLMs), including GPT-1, GPT-2, GPT-3/GPT-3.5 and GPT-4. **GPT-1** [23], introduced in 2018, was the initial variant of the Generative Pre-trained Transformer, which included 117 million parameters. In 2019, **GPT-2** [24] was released, boasting 1.5 billion parameters and delivering impressive language-generation capabilities. **GPT-3** [25], released in 2020, took a significant leap forward with staggering 175 billion parameters. The latest is **GPT-4** [26], which was released in 2023 estimated 100 trillion parameters. One notable enhancement in GPT-4 is that it is a multimodal model that can process images and text. GPT models have garnered significant attention and showcased remarkable advancements in natural-language processing, demonstrating their effectiveness across various applications.

8. EVALUATION METHODS

This section summarizes some famous evaluation metrics employed in deep learning-based Arabic QA systems. As indicated by [115], [131]-[132], extractive QA systems commonly utilize span-style datasets employing metrics like F1 score and Exact Match (EM). In contrast, ranking-based QA systems employ metrics such as C@1, Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR). On the other hand, abstractive QA systems, which rely on text and embedding-based methods, employ metrics like ROUGE, BLEU and several others like Word Mover's Distance (WMD) [133], Sentence Mover's Similarity (SMS), [134], BERTScore, [135] and MoverScore [134]. We classified the evaluation metrics into three main groups: traditional evaluation metrics, ranking-based metrics and textual and embedding-based metrics.

8.1 Traditional Evaluation Metrics

Traditional evaluation metrics, such as Precision, Recall and Accuracy, are often used to develop NLP systems and answer modeling problems. F1-score and Exact Match are commonly employed in QA tasks, particularly in span-style datasets, like the Arabic extractive QA datasets proposed in [82], [84].

Precision (P), Recall (R) and F1-measure (F1): Precision measures the ratio of the number of tokens in the prediction that overlap with the correct answer to the total number of tokens in the prediction (number of correct answers /number of questions answered). Either take: *Correct*(q) is the set of elements that form the perfect answer for q and $Found(q)$ are those that the QA system returned, then the precision (of the system as regards q) is the fraction of correct responses. The formula of precision is given in Equation 1.

$$Precision = \frac{|Found(q) \cap Correct(q)|}{|Found(q)|} \quad (1)$$

Recall, for each question, there is an expected set of correct answers; these are called the gold standard answers. Recall is the fraction of the correct elements that the system found (number of correct answers/number of questions to be answered). It is computed as shown in Equation 2.

$$Recall = \frac{|Found(q) \cap Correct(q)|}{|Correct(q)|} \quad (2)$$

F1-measure (also called F-score or F1-score) is the harmonic mean of precision and recall. F1 has been generally reserved for evaluating span-based question answering [131]. F1-measure is computed for the words of each predicted answer and each golden answer; i.e., is the average overlap between the words of the predicted and golden answers for a given question. It captures a system's precision and recall capabilities in a single score. It is computed as shown in Equation 3.

$$F1 - measure = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

Accuracy (Acc): Let *CA* denote the number of questions that a system answered correctly and *TQ* denote the total number of questions in the evaluation dataset. Then, accuracy represents the percentage of questions that a system answered correctly. Accuracy is defined as in Equation 4:

$$Accuracy = \frac{CA}{TQ} \tag{4}$$

Exact Match (EM): The name says it all. Exact match (EM) measures the proportion of predictions that match any reference answers exactly at the token level [131]. Put another way, the prediction is counted as correct only when it matches any reference answers to the given question. If *CP* denotes the number of correct predictions and *TQ* denotes the total number of questions in the evaluation dataset. EM is defined as in Equation 5:

$$EM = \frac{CP}{TQ} \tag{5}$$

8.2 Ranking-based Metrics

The performance evaluation of the QA ranking problem is based on the following metrics:

c@1: "A simple measure to assess non-response": This measure is used for question-answering systems that suppose one correct answer for each query. The measure works by evaluating question-answering systems by giving them an option not to answer the question rather than forcing them to provide an incorrect answer [136]. In other words, that differentiates between a wrongly answered question and an unanswered question. Equation 6 defined by [137] represents C@1, where *TQ* is; total questions, *CA* is the number of correctly answered questions and *UQ* is the number of unanswered questions:

$$c@1 = \frac{1}{TQ}(CA + UQ\frac{QA}{TQ}) \tag{6}$$

Mean Average Precision (MAP): It is the mean of the AveP scores for a set of queries. It is defined as in Equation 7.

$$MAP = \frac{1}{TQ}\sum_{i=1}^{TQ} AveP_i \tag{7}$$

Average Precision (AveP): $AveP_i$ is the average precision of the i[th] question and is computed as shown in Equation 8.

$$AveP = \frac{1}{TQ}\sum_{n=1}^{TQ}\frac{n}{Rank_n} \tag{8}$$

where $rank_i$ is the rank of the nt[h] correct answer.

MRR-Mean Reciprocal Rank: The mean reciprocal rank (MRR) is a relative score that calculates each question's average or mean of the reciprocal ranks. The reciprocal rank of a question is the multiplicative inverse of the rank of the first correct answer. It is defined by Equation 9.

$$MRR = \frac{1}{TQ}\sum_{n=1}^{TQ}\frac{1}{Rank_n} \tag{9}$$

8.3 Textual and Embedding-based Metrics

We provide a summary of some textual and embedding-based metrics, such as ROUGE and BLEU. Moreover, it has been observed that many evaluation metrics are primarily designed for English, which can limit their applicability to other languages with unique grammatical structures, such as Arabic. To address this challenge, several studies have proposed alternative metrics tailored to Arabic-text generation, including those based on textual features [138] and embedding-based approaches [139]-[140]. These contributions aim to improve the performance of Arabic-text generation models by better accounting for the linguistic characteristics of the target language.

ROUGE: (Recall-Oriented Under-study for Gisting Evaluation): It consists of a set of measures proposed initially to evaluate automatic text summarization [141]. So far, it is the most popular automatic method for evaluating the content of a summary by comparing the tokens of the candidate and the reference; i.e., it counts the number of over-lapping units, such as n-gram, word-pairs and word-sequences between the system-generated summary to be evaluated and the ideal summaries created by humans. The available variants of ROUGE measures are **ROUGE-N** (N =1,2,3,4), ROUGE-L, ROUGE-W and ROUGE-S. However, in this work, we focus on the most popular ROUGE metrics in question-answering systems, which are ROUGE-N and ROUGE-L, that represent a comparison of texts at different granularities. ROUGE-N measures the ratio of the number of overlaps of unigrams/bigrams/trigrams/four − grams

(single tokens) between the generated text and the reference text to the total n-grams in the reference text. It is defined by Equation 10:

$$ROUGE - N = \frac{\sum_{sr \in ref\ erences} \sum_{n-gram \in s_r} Count_{match}(n-gram)}{\sum_{sr \in ref\ erences} \sum_{n-gram \in s_r} Count(n-gram)} \quad (10)$$

BLEU: (Bilingual Evaluation Understudy), a precision-based metric originally proposed for machine translation [142], computes the $n-gram$ overlap between the reference and the hypothesis. Bleu is generally used for generative question-answering system evaluations. Using the following steps, we calculate the precision of different values of *n*. First, we calculate $Count_{clip}$ for any $n-gram$ as shown in Equation 11. Then, we calculate the modified precision score ($precision_n$), as shown in Equation 12.

$$Count_{clip} = \min(Count, Max\_Ref\_Count) \quad (11)$$

$$precision_n = \frac{\sum_{C \in \{C\ andidates\}} \sum_{garm_n \in C} Count_{clip}(n-gram)}{\sum_{C' \in \{Candidates\}} \sum_{garm_n' \in C'} Count_{clip}(n-gram')} \quad (12)$$

We add a brevity penalty to handle too short translations. BP is an exponential decay and is calculated as shown in Equation 13.

$$BrevityPenalty(BP) = \begin{cases} 1 & if\ c > r \\ e^{(1-\frac{r}{c})} & if\ c \leq r \end{cases} \quad (13)$$

With *r* being count of words in a reference translation and *c* count of words in a candidate translation. Finally, BLEU is defined as shown in Equation 14 with *N* no. of $n-grams$ (unigram, bigram, 3-gram, 4-gram); W_n denotes weight for each modified precision.

$$BLEU = BP.\exp(\sum_{n=1}^{N} W_n \log P_n) \quad (14)$$

AL-BLEU [139]: Traditional metrics like BLEU may not accurately assess the quality of Arabic translations due to the unique linguistic features of the language. To address this issue, Bouamor et al. introduced AL-BLEU (Arabic Language BLEU), a customized metric specifically designed for evaluating the quality of machine-translation (MT) systems in the context of Arabic-language translation. This metric leverages a human judgment corpus, where bilingual experts assess the quality of translations generated by various MT systems. The judgments are then used to calculate the AL-BLEU score, an adapted version of the original BLEU score. By considering the unique aspects of the Arabic language, AL-BLEU provides a more accurate assessment of translation quality for Arabic texts. The authors likely performed experimental comparisons between AL-BLEU and established metrics, such as BLEU and METEOR, to demonstrate the superiority of AL-BLEU in ranking MT systems. The results indicate that AL-BLEU highly correlates with human judgments and outperforms the other metrics.

Morphologically-enriched Embedding Metrics [140]: In the same context, Guzman et al. proposed an approach to evaluate MT in Arabic by using morphologically-enriched embeddings. The proposed method combines word-level and morpheme-level embeddings to capture the complexities of Arabic grammar and syntax [68]. This approach aims to address the limitations of conventional metrics and provide a more comprehensive assessment of translation quality in the context of Arabic intricate morphological structure, which can facilitate better communication and understanding between Arabic speakers and non-native speakers.

Automated Error Analysis [138]: It is a crucial aspect of natural-language processing (NLP) that helps researchers and developers evaluate the performance of their systems. El Kholy and Habash developed AMEANA, which is open-source and specifically designed to identify morphological errors in the output of MT systems compared to a gold standard reference. AMEANA provides detailed statistical reports on morphological errors and generates an altered version of the production that can be used to assess the impact of these errors using various evaluation metrics.

9. DISCUSSION

The increasing interest in developing an Arabic generative conversational AI system is due to the vital role of the Arab world in the global economy and politics. Consequently, there is a pressing need to address the challenges faced in conversational AI to create valuable resources and effective systems.

Notably, alongside the general challenges encountered in conversational AI, the specific challenges related to Arabic include:

- Limited resources: Arabic is a low-resource language, which means that there is a need for high-quality, annotated datasets for training conversational AI models. This scarcity of resources hinders the development of robust and accurate Arabic-language models [82]-[83], [111].
- Diverse Arabic dialects: Arabic has numerous dialects across different regions, each with its vocabulary, pronunciation and grammar. This diversity challenges building conversational AI systems that understand and generate responses in different Arabic dialects [68], [78].
- The complexity of Arabic script and grammar: Arabic has a rich and complex script with many characters and diacritical marks. The morphology and syntax of the language also add to its complexity. These factors make it challenging to process and analyze Arabic text accurately [68].

Advanced machine-learning algorithms, natural-language understanding models and language-specific libraries and tools are required to overcome these challenges. The development of Arabic AI technologies, such as Farasa [92] and CAMeL [93] Tools, to handle the morphology orthography and ambiguity in Arabic language can facilitate research in building Arabic conversational AI. There is also a need for more information, tips and insight to create chatbots that can converse in Arabic with the accuracy and technology of Arabic natural-language understanding improving daily. However, there are still challenges in creating and maintaining Arabic chatbots, compounded by a shortage in skills. Future research should concentrate on building efficient Arabic conversational systems using both word embeddings and ML methods to overcome the challenges of developing Arabic-language bots, such as the need for more dialectal task-oriented dialogue datasets.

Previous research on Arabic conversational systems has primarily focused on Modern Standard Arabic (MSA) and overlooked some local dialects [8], [75]. MSA is morphologically rich and complex compared to English, but easier to parse than Arabic dialects. Rule-based and hand-crafted feature-based methods form the foundation of NLP techniques for Arabic-language dialogue. However, these methods have limitations due to the complex nature of Arabic morphology orthographic variations, dialectal differences and a high degree of ambiguity. In contrast to English and other chatbots, Arabic chatbots still require further enhancement to become more robust and efficient. Moreover, recently, researchers have exploited the potential of chatGPT, such as Siu et al. [143] who mentioned that GPT models have limited capabilities for low-resource languages such as Arabic. Khoshafah [144], on the other hand, focuses on ChatGPT application in Arabic-English translation and advises users against relying solely on it for translations, recommending the involvement of a professional translator for more accurate results.

Several areas require attention and development for the future of Arabic conversational AI. Firstly, deep-learning techniques have shown promise in achieving high performance for conversational AI systems in English. However, constructing a large dataset for different question types in Arabic is necessary to achieve similar results. Secondly, developing solid tools, such as Farasa [92] and CAMeL Tools [93], to facilitate resolving Arabic morphology orthographic variations, dialectal differences and ambiguity, can improve the accuracy of Arabic conversational AI. Thirdly, improving publicly available pre-trained language models' (PTLMs) scalability, such as AraBERT [101] and ARBERT [145], by augmenting the training data, can enhance the performance of Arabic Conversational AI. Scaling the model's size or the amount of training data improves model capacity for downstream tasks [146]-[147]. Finally, focusing on building Arabic conversational AI on domains, such as criminal law, intellectual property and tax law texts, can enhance the accuracy and relevance of the responses. These developments can bridge the gap in dialectal speech recognition and overcome the challenges of limited resources, lack of standard orthographic rules and lack of definition in Arabic dialects [61][65][68].

10. Conclusion and Future Work

This paper reviews significant works on Arabic conversational systems. We found that these systems have received increasing attention from the NLP research community. We noticed that Arabic conversational agents commonly use rule-based and hand-crafted feature-based methods. At the same time, deep-learning techniques extensively explored in English dialogue systems require further improvement for Arabic. We

explore and discuss the challenges of the Arabic language and the proposed solutions in the literature. Three main evaluation methods for question-answering systems and conversational agents are identified: traditional evaluation metrics, ranking-based metrics and textual and embedding-based metrics. We found that deep-learning techniques show great promise in addressing the challenges of Arabic dialogue systems. However, constructing an extensive dataset encompassing various question types specific to the Arabic language is crucial for leveraging the potential of deep learning in Arabic dialogue systems. We highlight the significance of creating tools, such as MADAMIRA, Farasa and CAMeL Tools. These tools address the challenges of Arabic morphology orthographic complexities and ambiguity, ultimately supporting the development of Arabic conversational AI research.

As future work, we plan to create an efficient Arabic conversational system by leveraging contextualized word embedding and deep-learning methods. This approach aims to overcome challenges in creating Arabic-language bots. We also propose exploring Arabic spoken or written dialectical conversational agents, which researchers have not extensively investigated yet. Furthermore, we plan to improve publicly available PTLMs scalability, like ARBERT, by augmenting the training data. As shown by Kaplan et al. [146] and Zhao et al. [147], increasing the model size or the amount of training data can significantly improve the model performance on downstream tasks. Additionally, we aim to develop specialized Arabic question-answering systems tailored to specific domains, such as the judiciary and legal sector. We hope that this survey can provide researchers with a comprehensive overview of the current state of Arabic conversational systems, encompassing advancements in technique, employed approaches and outstanding challenges to facilitate progress and innovation within this field.

REFERENCES

[1] M. Adam, M. Wessel and A. Benlian, "Ai-based Chatbots in Customer Service and their Effects on User Compliance," Electronic Markets, vol. 31, no. 2, pp. 427–445, 2021.

[2] L. T. Car, D. A. Dhinagaran, B. M. Kyaw et al., "Conversational Agents in Health Care: Scoping Review and Conceptual Analysis," Journal of Medical Internet Research, vol. 22, no. 8, p. e17158, 2020.

[3] M. M. Mariani, N. Hashemi and J. Wirtz, "Artificial Intelligence Empowered Conversational Agents: A Systematic Literature Review and Research Agenda," J. of Business Research, vol. 161, p. 113838, 2023.

[4] M. Hijjawi, Z. Bandar, K. Crockett and D. Mclean, "Arabchat: An Arabic Conversational Agent," Proc. of the 6th IEEE Int. Conf. on Computer Science and Information Technology (CSIT), pp. 227–237, 2014.

[5] M. Hijjawi, H. Qattous and O. Alsheiksalem, "Mobile Arabchat: An Arabic Mobile-based Conversational Agent," Int. J. Adv. Comput. Sci. Appl. (IJACSA), vol. 6, no. 10, 2015.

[6] M. Hijjawi, Z. Bandar and K. Crockett, "The Enhanced Arabchat: An Arabic Conversational Agent," Int. J. of Advanced Computer Science and Applications, vol. 7, no. 2, 2016.

[7] S. S. Aljameel, J. D. O'Shea, K. A. Crockett, A. Latham and M. Kaleem, "Development of an Arabic Conversational Intelligent Tutoring System for Education of Children with ASD," Proc. of the 2017 IEEE Int. Conf. on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), pp. 24–29, Annecy, France, 2017.

[8] D. Al-Ghadhban and N. Al-Twairesh, "Nabiha: An Arabic Dialect Chatbot," Int. J. of Advanced Computer Science and Applications, vol. 11, no. 3, DOI: 10.14569/IJACSA.2020.0110357, 2020.

[9] D. Baidoo-Anu and L. O. Ansah, "Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of Chatgpt in Promoting Teaching and Learning," SSRN 4337484, [Online], Available: https://dx.doi.org/10.2139/ssrn.4337484, 2023.

[10] E. Kasthuri and S. Balaji, "A Chatbot for Changing Lifestyle in Education," Proc. of the 2021 3rd IEEE Int. Conf. on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), pp. 1317–1322, Tirunelveli, India, 2021.

[11] M. A. Kuhail, B. Abu Shawar and R. Hammad, Trends, Applications and Challenges of Chatbot Technology, Advances in Web Technologies and Engineering, IGI Global, ISBN10: 1668462346, 2023.

[12] A. D Fadhil et al., "Ollobot-towards a Text-based Arabic Health Conversational Agent: Evaluation and Results," Proc. of the Int. Conf. on Recent Advances in Natural Language Processing (RANLP 2019), pp. 295–303, Varna, Bulgaria, 2019.

[13] M. Boussakssou, H. Ezzikouri and M. Erritali, "Chatbot in Arabic Language Using Seq to Seq Model," Multimedia Tools and Applications, vol. 81, no. 2, pp. 2859–2871, 2022.

[14] L. Xu, L. Sanders, K. Li et al., "Chatbot for Health Care and Oncology Applications Using Artificial Intelligence and Machine Learning: Systematic Review," JMIR Cancer, vol. 7, no. 4, p. e27850, 2021.

[15] L. Athota, V. K. Shukla, N. Pandey and A. Rana, "Chatbot for Healthcare System Using Artificial Intelligence," Proc. of the 2020 IEEE 8th Int. Conf. on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), pp. 619–622, Noida, India, 2020.

[16] A. Stöckl, "Classification of Chatbot Inputs," [Online], Available: https://www.researchgate.net/publication/ 318661551_Classification_of_Chatbot_Inputs, 2017.
[17] N. K. Manaswi and S. John, Deep Learning with Applications Using Python, ISBN-10: 1484240510, Springer, 2018.
[18] N. M. Radziwill and M. C. Benton, "Evaluating Quality of Chatbots and Intelligent Conversational Agents," arXiv preprint, arXiv: 1704.04579, 2017.
[19] Amazon Lab126, "Amazon Alexa," [Online], Available: https://en.wikipedia.org/wiki/Amazon_Alexa/, 2013.
[20] S. Alhumoud et al., "Rahhal: A Tourist Arabic Chatbot," Proc. of the 2022 2nd IEEE Int. Conf. of Smart Systems and Emerging Technologies (SMARTTECH), pp. 66–73, 2022.
[21] Al-H. Al-Ajmi and N. Al-Twairesh, "Building an Arabic Flight Booking Dialogue System Using a Hybrid Rule-based and Data Driven Approach," IEEE Access, vol. 9, pp. 7043–7053, 2021.
[22] A. M. Rahman, A. Al Mamun and A. Islam, "Programming Challenges of Chatbot: Current and Future Prospective," Proc. of the 2017 IEEE Region 10 Humanitarian Technology Conf. (R10-HTC), pp. 75–78, Dhaka, Bangladesh, 2017.
[23] A. Radford, K. Narasimhan, T. Salimans et al., "Improving Language Understanding by Generative Pre-training," pp. 1-12, [Online], Available: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf, 2018.
[24] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever et al., "Language Models Are Unsupervised Multitask Learners," OpenAI blog, vol. 1, no. 8, p. 9, 2019.
[25] T. Brown, B. Mann, N. Ryder et al., "Language Models are Few-shot Learners," Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020.
[26] OpenAI, "Gpt-4 Technical Report," [Online], Available: https://cdn.openai.com/papers/gpt-4.pdf, 2023.
[27] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint, arXiv: 1810.04805, 2018.
[28] Romal Thoppilan et al., "LaMDA: Language Models for Dialog Applications," arXiv preprint, arXiv: 2201.08239, 2022.
[29] P. Brereton, B. A Kitchenham, D. Budgen, M. Turner and M. Khalil, "Lessons from Applying the Systematic Literature Review Process within the Software Engineering Domain," Journal of Systems and Software, vol. 80, no. 4, pp. 571–583, 2007.
[30] Y. Xiao and M. Watson, "Guidance on Conducting a Systematic Literature Review," Journal of Planning Education and Research, vol. 39, no. 1, pp. 93–112, 2019.
[31] A. M. Turing, Computing Machinery and Intelligence, in Book: Parsing the Turing Test, pp. 23–65, Springer, 2009.
[32] J. Weizenbaum et al., "Eliza: A Computer Program for the Study of Natural Language Communication between Man and Machine," Communications of the ACM, vol. 9, no. 1, pp. 36–45, 1966.
[33] E. Adamopoulou and L. Moussiades, "Chatbots: History, Technology and Applications," Machine Learning with Applications, vol. 2, p. 100006, 2020.
[34] Loebner Prize, [Online], Available: https://en.wikipedia.org/wiki/Loebner_Prize.
[35] Ina, [Online], Available: https://onlim.com/en/the-history-of-chatbots/, 2021.
[36] R. Wallace, Artificial Linguistic Internet Computer Entity (Alice), [Online], Available: https://www.chatbots.org/chatbot/a.l.i.c.e/, 1995.
[37] B. Abu Shawar and E. Atwell, "Chatbots: Are They Really Useful?" J. for Language Technology and Computational Linguistics, vol. 22, no. 1, pp. 29–49, 2007.
[38] A. M. Ezzeldin and M. Shaheen, "A Survey of Arabic Question Answering: Challenges, Tasks, Approaches, Tools and Future Trends," Proc. of the 13th Int. Arab Conf. on Information Technology (ACIT 2012), pp. 1–8, 2012.
[39] Y. Wu, G. Wang, W. Li and Z. Li, "Automatic Chatbot Knowledge Acquisition from Online Forum *via* Rough Set and Ensemble Learning," Proc. of the 2008 IEEE IFIP Int. Conf. on Network and Parallel Computing, pp. 242–246, Shanghai, China, 2008.
[40] G. Molnár and Z. Szüts, "The Role of Chatbots in Formal Education," Proc. of the 2018 IEEE 16th Int. Symposium on Intelligent Systems and Informatics (SISY), pp. 000197–000202, Subotica, Serbia, 2018.
[41] S. Worswick, Mitsuku Chatbot, [Online], Available: https://www.pandorabots.com/mitsuku/, 2005.
[42] David Ferrucci, IBM Watson, [Online], Available: https://www.ibm.com/watson, 2006.
[43] T. G. D. Kittlaus and UCLA Alumnus Adam Cheyer, Apple Siri, [Online], Available: https://www.apple.com/siri/, 2010.
[44] Google, Google Assistant, [Online], Available: https://assistant.google.com/, 2012.
[45] Google Cloud, Dialogflow, [Online], Available: https://cloud.google.com/dialogflow, 2016.
[46] Microsoft Azure, Language Understanding (LUIS), [Online], Available: https://www.luis.ai/, 2017.
[47] Amazon, Amazon Lex- Conversational AI for Chatbots, [Online], Available: https://aws.amazon.com/lex/?nc=sn&loc=0, 2017.
[48] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention Is All You Need," Advances in Neural Information

Processing Systems, vol. 30, arXiv:1706.03762, 2017.
[49] S. Hussain, O. A. Sianaki and N. Ababneh, "A Survey on Conversational Agents/Chatbots Classification and Design Techniques, Proc. of the Workshops of the Int. Conf. on Advanced Information Networking and Applications, pp. 946–956, Springer, 2019.
[50] E. Adamopoulou and L. Moussiades, "An Overview of Chatbot Technology," Proc. of the IFIP Int. Conf. on Artificial Intelligence Applications and Innovations, pp. 373–383, Springer, 2020.
[51] E. H. Almansor and F.h K. Hussain, "Survey on Intelligent Chatbots: State-of-the-art and Future Research Directions," Proc. of Conf. on Complex, Intelligent and Software Intensive Systems, pp. 534–543, Springer, 2019.
[52] R. Dale, "The Return of the Chatbots," Natural Language Engineering, vol. 22, no. 5, pp. 811–817, 2016.
[53] M. B. Hoy, "Alexa, Siri, Cortana and More: An Introduction to Voice Assistants," Medical Reference Services Quarterly, vol. 37, no. 1, pp. 81–88, 2018.
[54] A. L. Guzman, "Making AI Safe for Humans: A Conversation with Siri," Chapter in Book: Socialbots and Their Friends, 1st Edn., pp. 85–101, Routledge, 2016.
[55] K. Nimavat and T. Champaneria, "Chatbots: An Overview of Types, Architecture, Tools and Future Possibilities," Int. J. Sci. Res. Dev. (IJSRD), vol. 5, no. 7, pp. 1019–1024, 2017.
[56] H. T. Hien, P.-N. Cuong, L. N. H. Nam, H. L. T. K. Nhung and L. D. Thang, "Intelligent Assistants in Higher-education Environments: The FIT-EBoT, a Chatbot for Administrative and Learning Support," Proc. of the 9th Int. Symposium on Inform. and Comm. Techn. (SoICT '18), pp. 69–76, 2018.
[57] K. Ramesh, S. Ravishankaran, A. Joshi and K. Chandrasekaran, "A Survey of Design Techniques for Conversational Agents," Proc. of the Int. Conf. on Information, Communication and Computing Technology, pp. 336–350, Springer, 2017.
[58] Z. Ji, Z. Lu and H. Li, "An Information Retrieval Approach to Short Text Conversation," arXiv preprint, arXiv: 1408.6988, 2014.
[59] R. Artstein, S. Gandhe, J. Gerten, A. Leuski and D. Traum, "Semi-formal Evaluation of Conversational Characters," in Book: Languages: From Formal to Natural, vol. 5533, pp. 22–35, Springer, 2009.
[60] E. Atwell, "A Chatbot As a Question Answering Tool," DOI: 10.17758/ur.u0915120, 2015.
[61] R. Artstein, A. Gainer, K. Georgila et al., "New Dimensions in Testimony Demonstration," Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 32–36, San Diego, California, USA, 2016.
[62] R. Artstein, A. Leuski, H. Maio et al., "How Many Utterances Are Needed to Support Time-offset Interaction?" Proc. of the 28th Int. Florida Artificial Intelligence Research Society Conference (FLAIRS 2015), pp. 144-149, 2015.
[63] D. Traum, K. Georgila, R. Artstein and A. Leuski, "Evaluating Spoken Dialogue Processing for Time-offset Interaction," Proc. of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 199–208, Prague, Czech Republic, 2015.
[64] D. Traum et al., "New Dimensions in Testimony: Digitally Preserving a Holocaust Survivors' Interactive Storytelling," Proc. of the 8th Int. Conf. on Interactive Digital Storytelling (ICIDS 2015), Copenhagen, Denmark, Proceedings 8, pp. 269–281, Springer, 2015.
[65] D. Abu Ali et al., "A Bilingual Interactive Human Avatar Dialogue System," Proc. of the 19th Annual SIGdial Meeting on Discourse and Dialogue, pp. 241–244, Melbourne, Australia, 2018.
[66] A. Farghaly and K. Shaalan, "Arabic Natural Language Processing: Challenges and Solutions," ACM Transactions on Asian Language Information Processing (TALIP), vol. 8, no. 4, pp. 1–22, 2009.
[67] K. C. Ryding, A Reference Grammar of Modern Standard Arabic, ISBN: 0521777712, Cambridge University Press, 2005.
[68] N. Y. Habash, Introduction to Arabic Natural Language Processing, ISBN: 1598297953, Springer Nature, 2022.
[69] C. Zhai, "A Systematic Review on Artificial Intelligence Dialogue Systems for Enhancing English As Foreign Language Students' Interactional Competence in the University," Computers and Education: Artificial Intelligence, vol. 4, p. 100134, 2023.
[70] K. Chemnad and A. Othman, "Advancements in Arabic Text-to-speech Systems: A 22-year Literature Review," IEEE Access, vol. 11, pp. 30929 – 30954, 2023.
[71] K. Darwish, N. Habash, M. Abbas et al., "A Panoramic Survey of Natural Language Processing in the Arab World," Communications of the ACM, vol. 64, no. 4, pp. 72–81, 2021.
[72] Amira Dhouib et al., "Arabic Automatic Speech Recognition: A Systematic Literature Review," Applied Sciences, vol. 12, no. 17, p. 8898, 2022.
[73] M. Hijjawi and Y. Elsheikh, "Arabic Language Challenges in Text Based Conversational Agents Compared to the English Language," IJCSIT, vol. 7, no. 5, pp. 1–13, 2015.
[74] N. Habash, M. T. Diab and O. Rambow, "Conventional Orthography for Dialectal Arabic," Proc. of the 8th Int. Conf. on Language Resources and Evaluation (LREC'12), pp. 711–718, Istanbul, Turkey, 2012.
[75] D. Abu Ali and N. Habash, "Botta: An Arabic Dialect Chabot," Proc. of COLING 2016, the 26th Int. Conf. on Computational Linguistics: System Demonstrations, pp. 208–212, Osaka, Japan, 2016.

[76] W. Zaghouani et al., "Large Scale Arabic Error Annotation: Guidelines and Framework," Proc. of the 9th Int. Conf. on Language Resources and Evaluation (LREC'14), pp. 2362-2369, Reykjavik, Iceland, 2014.

[77] R. Eskander, N. Habash, O. Rambow and N. Tomeh, "Processing Spontaneous Orthography," Proc. of the 2013 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 585–595, Atlanta, USA, 2013.

[78] N. Habash, R. Roth, O. Rambow, R. Eskander and N. Tomeh, "Morphological Analysis and Disambiguation for Dialectal Arabic," Proc. of the 2013 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Lang. Techn., pp. 426–432, Atlanta, USA, 2013.

[79] N. Habash, A. Soudi and T. Buckwalter, "On Arabic Transliteration," in Chapter: Arabic Computational Morphology: Knowledge-based and Empirical Methods, Part of the Text, Speech and Language Technology Book Series, vol. 38, pp. 15–22, 2007.

[80] T. Buckwalter, "Buckwalter Arabic Morphological Analyzer Version 1.0," Catalog no. LDC2002L49, Linguistic Data Consortium, University of Pennsylvania, 2002.

[81] T. H. Alwaneen et al., "Arabic Question Answering System: A Survey," Artificial Intelligence Review, vol. 55, no. 1, pp. 207–253, 2022.

[82] H. Mozannar, K. El Hajal, E. Maamary and H. Hajj, "Neural Arabic Question Answering," arXiv preprint, arXiv: 1906.05394, 2019.

[83] W. Antoun, F. Baly and H. Hajj, "Araelectra: Pre-training Text Discriminators for Arabic Language Understanding," arXiv preprint, arXiv: 2012.15516, 2020.

[84] Jonathan H Clark et al., "TyDi QA: A Benchmark for Information-seeking Question Answering in Typologically Diverse Languages," Transactions of the Association for Computational Linguistics, vol. 8, pp. 454–470, 2020.

[85] B. Abu Shawar, "A Chatbot As a Natural Web Interface to Arabic Web QA," Int. J. of Emerging Technologies in Learning (iJET), vol. 6, no. 1, pp. 37–43, 2011.

[86] M. A. Yaghan, ""Arabizi": A Contemporary Style of Arabic Slang," Design Issues, vol. 24, no. 2, pp. 39– 52, 2008.

[87] K. Shaalan, "A Survey of Arabic Named Entity Recognition and Classification," Computational Linguistics, vol. 40, no. 2, pp. 469–510, 2014.

[88] W. Bakari, P. Bellot and M. Neji, "Researches and Reviews in Arabic Question Answering: Principal Approaches and Systems with Classification," Proc. of the Int. Arab Conf. on Information Technology (ACIT'2016), pp. 1–9, 2016.

[89] Arfath Pasha et al., "MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic," Proc. of the 9th Int. Conf. on Language Resources and Evaluation (LREC'14), pp. 1094–1101, Reykjavik, Iceland, 2014.

[90] N. Habash and O. Rambow, "Arabic Tokenization, Part-of-speech Tagging and Morphological Disambiguation in One Fell Swoop," Proc. of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), pp. 573–580, Ann Arbor, Michigan, USA, 2005.

[91] N. Habash, O. Rambow and R. Roth, "MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization," Proc. of the 2nd Int. Conf. on Arabic Language Resources and Tools (MEDAR), vol. 41, p. 62, Cairo, Egypt, 2009.

[92] A. Abdelali, K. Darwish, N. Durrani and H. Mubarak, "Farasa: A Fast and Furious Segmenter for Arabic," Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 11–16, San Diego, California, USA, 2016.

[93] O. Obeid et al., "CAMeL Tools: An Open Source Python Toolkit for Arabic Natural Language Processing," Proc. of the 12th Language Resources and Evaluation Conference, pp. 7022–7032, Marseille, France, 2020.

[94] Zaid Alyafeai et al., "Masader: Metadata Sourcing for Arabic Text and Speech Data Resources," arXiv preprint, arXiv: 2110.06744, 2021.

[95] N. Al-Twairesh, H. Al-Khalifa, A. Alsalman and Y. Al-Ohali, "Sentiment Analysis of Arabic Tweets: Feature Engineering and a Hybrid Approach," arXiv preprint, arXiv: 1805.08533, 2018.

[96] P. Liu, X. Qiu, J. Chen and X.-J. Huang, "Deep Fusion LSTMs for Text Semantic Matching," Proc. of the 54th Annual Meeting of the Association for Computational Linguistics, vol. 1: Long Papers, pp. 1034–1043, Berlin, Germany, 2016.

[97] J. Mueller and A. Thyagarajan, "Siamese Recurrent Architectures for Learning Sentence Similarity," Proc. of the AAAI Conf. on Artificial Intelligence, vol. 30, no. 1, DOI: 10.1609/aaai.v30i1.10350, 2016.

[98] N. Van Tu et al., "A Deep Learning Model of Multiple Knowledge Sources Integration for Community Question Answering," VNU J. of Science: Computer Sci. and Comm. Eng., vol. 37, no. 1, 2021.

[99] J. D. Ming-Wei C. Kenton and L. K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proc. of NAACL-HLT, pp. 4171– 4186, Minneapolis, USA, 2019.

[100] A. Hamza, N. En-Nahnahi and S. El Alaoui Ouatik, "Exploring Contextual Word Representation for Arabic Question Classification," Proc. of the 2020 1st IEEE Int. Conf. on Innovative Research in Applied Science, Engineering and Technology (IRASET), pp. 1–5, Meknes, Morocco, 2020.

[101] W. Antoun, F. Baly and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," arXiv preprint, arXiv: 2003.00104, 2020.
[102] A. Almiman, N. Osman and M. Torki, "Deep Neural Network Approach for Arabic Community Question Answering," Alexandria Engineering Journal, vol. 59, no. 6, pp. 4427–4434, 2020.
[103] D. Elalfy, W. Gad and R. Ismail, "A Hybrid Model to Predict Best Answers in Question Answering Communities," Egyptian Informatics Journal, vol. 19, no. 1, pp. 21–31, 2018.
[104] D. Elalfy, W. Gad and R. Ismail, "Predicting Best Answer in Community Questions Based on Content and Sentiment Analysis," Proc. of the 2015 IEEE 7th Int. Conf. on Intelligent Computing and Information Systems (ICICIS), pp. 585–590, Cairo, Egypt, 2015.
[105] A. Hamza et al., "An Arabic Question Classification Method Based on New Taxonomy and Continuous Distributed Representation of Words," Journal of King Saud University-Computer and Information Sciences, vol. 33, no. 2, pp. 218–224, 2021.
[106] N. Othman, R. Faiz and K. Smaïli, "Learning English and Arabic Question Similarity with Siamese Neural Networks in Community Question Answering Services," Data & Knowledge Engineering, vol. 138, p. 101962, 2022.
[107] W.-N. Zhang et al., "Capturing the Semantics of Key Phrases Using Multiple Languages for Question Retrieval," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 4, pp. 888–900, 2015.
[108] W. Bakari and M. Neji, "A Novel Semantic and Logical-based Approach Integrating RTE Technique in the Arabic Question–answering," International Journal of Speech Technology, vol. 25, pp. 1–17, 2020.
[109] W. Bakari, P. Bellot and M. Neji, "AQA-WebCorp: Web-based Factual Questions for Arabic," Procedia Computer Science, vol. 96, pp. 275–284, 2016.
[110] N. A. Al-Madi et al., "An Intelligent Arabic Chatbot System Proposed Framework," Proc. of the 2021 IEEE Int. Conf.on Information Technology (ICIT), pp. 592–597, Amman, Jordan, 2021.
[111] K. Alsubhi, A. Jamal and A. Alhothali, "Deep Learning-based Approach for Arabic Open Domain Question Answering," PeerJ Computer Science, vol. 8, p. e952, 2022.
[112] A. Soliman, K Eissa and S. R El-Beltagy, "Aravec: A Set of Arabic Word Embedding Models for Use in Arabic NLP," Procedia Computer Science, vol. 117, pp. 256–265, 2017.
[113] M. Ben-Sghaier, W. Bakari and M. Neji, "An Arabic Question-answering System Combining a Semantic and Logical Representation of Texts," Proc. of the Int. Conf. on Intelligent Systems Design and Applications, pp. 735–744, Springer, 2017.
[114] V. Karpukhin et al., "Dense Passage Retrieval for Open-domain Question Answering," arXiv preprint, arXiv: 2004.04906, 2020.
[115] H. Abdel-Nabi, A. Awajan and M. Z. Ali, "Deep Learning-based Question Answering: A Survey," Knowledge and Information Systems, vol. 65, no. 4, pp. 1399–1485, 2023.
[116] S. T. Chung and R. L. Morris, "Isolation and Characterization of Plasmid Deoxyribonucleic Acid from Streptomyces Fradiae," Paper presented at the 3rd Int. Symposium on the Genetics of Industrial Microorganisms, University of Wisconsin, Madison, 4–9 June 1978.
[117] Z. Hao, A. AghaKouchak, N. Nakhjiri and A. Farahmand, "Global Integrated Drought Monitoring and Prediction System," Scientific Data, vol. 1, p. 853801, 2014.
[118] S. A. Babichev, J. Ries and A. I. Lvovsky, "Quantum Scissors: Teleportation of Single-mode Optical States by Means of a Nonlocal Single Photon," arXiv preprint, arXiv: 0208066v1, 2002.
[119] M. Beneke, G. Buchalla and I. Dunietz, "Mixing Induced CP Asymmetries in Inclusive B Decays," Physics Letters, B393, pp. 132–142, 1997.
[120] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, "Enriching Word Vectors with Subword Information," Transactions of the Association for Computational Linguistics, vol. 5, pp. 135– 146, 2017.
[121] B. Stahl, "DeepSIP: Deep Learning of Supernova Ia Parameters," Astrophysics Source Code Library, Bibcode: 2020ascl.soft06023S, 2020.
[122] R. Yan, Y. Song and H. Wu, "Learning to Respond with Deep Neural Networks for Retrieval-based Human-computer Conversation System," Proc. of the 39th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR '16), pp. 55–64, 2016.
[123] Y. Wu, W. Wu, C. Xing, M. Zhou and Z. Li, "Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots," arXiv preprint, arXiv: 1612.01627, 2016.
[124] M. Wang, Z. Lu, Hang Li and Q. Liu, "Syntax-based Deep Matching of Short Texts," Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI 2015), pp. 1354-1361, 2015.
[125] I. Serban, A. Sordoni, Y. Bengio, A. Courville and J. Pineau, "Building End-to-end Dialogue Systems Using Generative Hierarchical Neural Network Models," Proc. of the AAAI Conf. on Artificial Intelligence, vol. 30, pp. 3776–3783, 2016.
[126] W. Zhang et al., "Context-sensitive Generation of Open-domain Conversational Responses," Proc. of the 27th Int. Conf. on Computational Linguistics, pp. 2437–2447, Santa Fe, New Mexico, USA, 2018.
[127] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai and X. Huang, "Pre-trained Models for Natural Language Processing: A Survey," Science China Technological Sciences, vol. 63, no. 10, pp. 1872–1897, 2020.
[128] K. S. Kalyan, A. Rajasekharan and S. Sangeetha, "AMMUS: A Survey of Transformer-based Pre-trained

Models in Natural Language Processing," arXiv preprint, arXiv: 2108.05542, 2021.
[129] M. E. Peters et al., "Deep Contextualized Word Representations," Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers), pp. 2227–2237, New Orleans, USA, 2018.
[130] C. Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-text Transformer," The Journal of Machine Learning Research, vol. 21, no. 1, pp. 5485–5551, 2020.
[131] P. Rajpurkar, J. Zhang, K. Lopyrev and P Liang, "Squad: 100,000+ Questions for Machine Comprehension of Text," arXiv preprint, arXiv: 1606.05250, 2016.
[132] A. Celikyilmaz, E. Clark and J. Gao, "Evaluation of Text Generation: A Survey," arXiv preprint, arXiv: 2006.14799, 2020.
[133] M. Kusner, Y. Sun, N. Kolkin and K. Weinberger, "From Word Embeddings to Document Distances," Proc. of the Int. Conf. on Machine Learning, PMLR, vol. 37, pp. 957-966, 2015.
[134] E. Clark, A. Celikyilmaz and N. A. Smith, "Sentence Mover's Similarity: Automatic Evaluation for Multi-sentence Texts," Proc. of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 2748–2760, Florence, Italy, 2019.
[135] W. Zhao et al., "MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover Distance," arXiv preprint, arXiv: 1909.02622, 2019.
[136] P. Forner et al., "Evaluating Multilingual Question Answering Systems at CLEF," Proc. of the 7th Int. Conf. on Language Resources and Evaluation (LREC'10), Valletta, Malta, 2010.
[137] A. Peñas and A. Rodrigo, "A Simple Measure to Assess Non-response," Proc. of the 49th Annual Meeting of the Association for Computational Linguistics, pp. 1415–1424, Portland Oregon, 2011.
[138] A. El Kholy and N. Habash, "Automatic Error Analysis for Morphologically Rich Languages," Proc. of Machine Translation Summit XIII: Papers, pp. 225-232, 2011.
[139] H. Bouamor, H. Alshikhabobakr, B. Mohit and K. Oflazer, "A Human Judgement Corpus and a Metric for Arabic MT Evaluation," Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP), pp. 207–213, Doha, Qatar, 2014.
[140] F. Guzmán, H. Bouamor, R. Baly and N. Habash, "Machine Translation Evaluation for Arabic Using Morphologically-enriched Embeddings," Proc. of COLING 2016, the 26th Int. Conf. on Computational Linguistics: Technical Papers, pp. 1398–1408, Osaka, Japan, 2016.
[141] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," Proc. of Text Summarization Branches Out, pp. 74–81, Association for Computational Linguistics, Barcelona, Spain, 2004.
[142] K. Papineni, S. Roukos, T. Ward and W.-J. Zhu, "BLEU: A Method for Automatic Evaluation of Machine Translation," Proc. of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318, Philadelphia, USA, 2002.
[143] S. C. Siu, "ChatGPT and GPT-4 for Professional Translators: Exploring the Potential of Large Language Models in Translation," SSRN 4448091, DOI: 10.2139/ssrn.4448091, 2023.
[144] F. Khoshafah, "ChatGPT for Arabic-English Translation: Evaluating the Accuracy," Research Square, DOI: 10.21203/rs.3.rs-2814154/v1, 2023.
[145] M. Abdul-Mageed, A. Elmadany and M. B. Nagoudi, "ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic," arXiv preprint, arXiv: 2101.01785, 2020.
[146] Jared Kaplan et al., "Scaling Laws for Neural Language Models," arXiv: 2001.08361, 2020.
[147] Wayne Xin Zhao et al., "A Survey of Large Language Models," arXiv preprint, arXiv: 2303.18223, 2023.

ملخص البحث:

تقدّم هذه الدّراسة عرضاً شاملاً للأعمال البحثية في هذا المجال؛ من أجْل بيان مزايا الأنظمة المتعلّقة بالمحادثة باللّغة العربية، الى جانب التّحدّيات التي تعترضها. يبدأ التّحليل بنظرةٍ على تاريخ أنظمة المحادثة وتصنيفها، ثمّ يتمّ الانتقال الى الصّعوبات والتّحدّيات التي تواجه تصميم أنظمة المحادثة باللّغة العربية باستخدام تقنيات الذّكاء الاصطناعي. كذلك تتطرّق الدّراسة الى ما حصل من تطوّر في أنظمة المحادثة باللّغة العربية بفضل التّقدُّم الّذي حدث في تقنيات التّعلُّم العميق. بالإضافة الى ما سبق، تستعرض الدّراسة مؤشّرات التّقييم الّتي تُستخدم في الحُكم على أنظمة المحادثة باللّغة العربية والمفاضلة بينها.

جامعة الأميرة سميّة للتكنولوجيا
Princess Sumaya University for Technology

صندوق دعم البحث العلمي والابتكار
Scientific Research and Innovation Support Fund

المجلة الأردنية للحاسوب وتكنولوجيا المعلومات

ISSN 2415 - 1076 (Online)
ISSN 2413 - 9351 (Print)

أيلول ٢٠٢٣ | المجلد ٩ | العدد ٣

الصفحات	عنوان البحث

JJCIT

www.jjcit.org

jjcit@psut.edu.jo

مجلة علمية عالمية متخصصة تصدر
بدعم من صندوق دعم البحث العلمي والابتكار