



Article

Arabic Tweets-Based Sentiment Analysis to Investigate the Impact of COVID-19 in KSA: A Deep Learning Approach

Arwa Alqarni and Atta Rahman *

Department of Computer Science, College of Computer Science and Information Technology,
Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia

* Correspondence: aaurrahman@iau.edu.sa

Abstract: The World Health Organization (WHO) declared the outbreak of Coronavirus disease 2019 (COVID-19) a pandemic on 11 March 2020. The evolution of this pandemic has raised global health concerns, making people worry about how to protect themselves and their families. This has greatly impacted people's sentiments. There was a dire need to investigate a large amount of social data such as tweets and others that emerged during the post-pandemic era for the assessment of people's sentiments. As a result, this study aims at Arabic tweet-based sentiment analysis considering the COVID-19 pandemic in Saudi Arabia. The datasets have been collected in two different periods in three major regions in Saudi Arabia, which are: Riyadh, Dammam, and Jeddah. Tweets were annotated with three sentiments: positive, negative, and neutral after due pre-processing. Convolutional neural networks (CNN) and bi-directional long short memory (BiLSTM) deep learning algorithms were applied for classifying the sentiment of Arabic tweets. This experiment showed that the performance of CNN achieved 92.80% accuracy. The performance of BiLSTM was scored at 91.99% in terms of accuracy. Moreover, as an outcome of this study, an overwhelming upsurge in negative sentiments were observed in the dataset during COVID-19 compared to the negative sentiments of the dataset before COVID-19. The technique has been compared with the state-of-the-art techniques in the literature and it was observed that the proposed technique is promising in terms of various performance parameters.

Keywords: COVID-19; sentiment analysis; Twitter; deep learning algorithm; CNN; BiLSTM



Citation: Alqarni, A.; Rahman, A. Arabic Tweets-Based Sentiment Analysis to Investigate the Impact of COVID-19 in KSA: A Deep Learning Approach. *Big Data Cogn. Comput.* **2023**, *7*, 16. <https://doi.org/10.3390/bdcc7010016>

Academic Editor: Min Chen

Received: 20 November 2022

Revised: 15 December 2022

Accepted: 11 January 2023

Published: 13 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In March 2020, the WHO classified the COVID-19 as a pandemic. COVID-19 is an infectious disease that assaults the respiratory system and attacks the lungs. The most common symptoms of COVID-19 are cough, fever, dyspnea, and viral pneumonia [1]. The COVID-19 epidemic has impacted all facets of human life and has grown into a socio-economic and psychological crisis [2]. As a result, governments have established several precautionary measures to halt the virus's outbreak. Precautionary measures included school closures, travel blocks, prohibition of social and public events, social distancing, and other measures. Consequently, people's daily routines have been impacted, and most social chatting and interactions have moved online through social networking sites such as Facebook, WhatsApp, Twitter, Zoom, and others. Usually, people have been using social media platforms to convey their feelings and emotions to the community. Furthermore, a lot of companies and organizations are using social media for marketing their products and services and studying feedback from customers to enhance their products and services, producing massive amounts of data. During the pandemic, social media was a strong tool that took on a crucial role where people could search for the most recent information and monitor what was happening in real-time [3]. However, with the rise of social media use, enormous volumes of important data have become easily accessible online. Accordingly, the demand for sentiment analysis during the pandemic that can study the users' sentiments and classify them has significantly increased.

Many natural language processing (NLP) researchers have concentrated on text classification such as sentiment analysis, topic detection, and language detection [4]. Sentiment Analysis (SA) is a kind of NLP task used with machine learning and/or deep learning techniques for classifying sentiments by subjective information. SA, known as an opinion mining, is the study and examination of individuals' attitudes, sentiments, assessments, and opinions about matters such as services, products, individuals, organizations, events, issues, elections, and subjects [5]. According to the Internet World Stats website, the Arabic language is ranked fourth among the ten most used languages on the internet [6]. Studies on sentiment analysis in the Arabic language are comparatively scarcer than those in the English language [7]. The Arabic language's complexity is most likely the reason for research limitations; one word can have many shapes employing numerous affixes, suffixes, and prefixes. The Arabic language can be divided into three groups: Classical Arabic (CA), Dialect Arabic (DA), and Modern Standard Arabic (MSA) [8]. CA can be found in religious books and scriptures. It is also mainly utilized in reading the Holy Qur'an and prayers. MSA can be employed in formal contexts such as newspaper articles, study rooms, and communication channels. Moving to the DA, it is utilized in everyday life discussions or posts on social media and is more diverse in nature. With the recent success of deep learning techniques in many sectors, researchers and data scientists have been widely using it to process the tasks of NLP including SA. Deep learning can handle complicated problems and process multiple layers to extract features from the input data [9]. Deep neural networks have also demonstrated their efficacy and efficiency on huge datasets [10].

This study seeks to implement an Arabic sentiment analysis to examine the impact on the opinions about COVID-19 in Saudi Arabia by applying deep learning techniques. The goal of this study has been accomplished by gathering an Arabic dataset from Twitter, using the Tweepy tool for two periods. The first period was during the COVID-19 pandemic, while the second period was before the COVID-19 pandemic. In this paper, we focused on the Saudi DA and MSA only. The dataset has been annotated into three classes: positive, negative, and neutral. Then, preprocessing and word embedding techniques were implemented to the dataset to prepare it for further analysis. To classify the sentiment of Arabic tweets, the deep learning algorithms, CNN and BiLSTM, have been applied. The results of the models are compared with each other and with other previous state-of-the-art studies in the literature. A word cloud for all datasets has been presented to show the most frequent words that appeared during the COVID-19 pandemic. Finally, the result of sentiment during the COVID-19 period has been compared with the result of sentiment before the COVID-19 period to present the impacts of COVID-19 in Saudi Arabia. In other words, this study attempted to gauge the effect of the COVID-19 epidemic on Saudi Arabia's population.

The major contributions of the study are:

1. Building an annotated dataset for Arabic (Saudi dialect and MSA) for sentiment analysis before and during the COVID-19 pandemic.
2. Building a suitable pre-trained Arabic word embedding for Arabic datasets applying deep learning algorithms.
3. Evaluating various deep learning algorithms for classifying the polarity of tweets and comparing it with the state-of-the-art techniques in the literature.
4. Identifying and comparing the Arabic tweets sentiment analysis before and during the COVID-19 pandemic in Saudi Arabia.

The rest of this paper is organized as follows: Section 2 gives a review of the literature. Section 3 presents the proposed methodology. Section 4 shows the experimental results and discussion. Lastly, we conclude this paper in Section 5.

2. Review of Literature

Social media sites such as Facebook and Twitter are regarded as sources of datasets that are frequently employed in applications of sentiment analysis. The Tweets usually present updated information on users' interests, feelings, behaviors, and opinions about global

events. During the COVID-19 epidemic, most individuals attentively used social media to keep fully informed and share their feelings about this epidemic. As a result, researchers were able to conduct studies on COVID-19 datasets collected from social networking sites for studying sentiment analysis during the pandemic. Many studies on sentiment analysis in English and Arabic have been published. Arabic sentiment analysis remains a major challenge, especially regarding the COVID-19 outbreak. However, traditional machine learning and deep learning methods are usually used for classifying sentiment analysis. Therefore, this literature review is organized into three sections, which are Arabic Sentiment Analysis using the Traditional Machine Learning Approaches, Arabic Sentiment Analysis using Deep Learning Approaches, and Sentiment Analysis Related to COVID-19.

2.1. Arabic Sentiment Analysis Using the Traditional Machine Learning Approaches

Several works in Arabic sentiment analysis used the machine learning method to study sentiment analysis. In [11], the authors studied Arabic sentiment analysis related to depression among Arabic tweeters. The data were gathered from Arabic tweeters who responded to a questionnaire from the Center for Epidemiologic Studies Depression Scale (CES-D). Furthermore, the dataset was manually annotated into three classes: depressed, non-depressed, and neutral. They used six machine learning techniques for Arabic tweet classification, which are SVM, AdaBoost, Random Forest (RF), Logistic Regression (LR), AdaBoost, and KNN algorithms. The findings demonstrate that the RF classification algorithm performed the best accuracy with 82.39% compared to the rest of the algorithms. In [12], the researchers applied machine learning algorithms for classifying the sentiment of Arabic reviews related to cafes and restaurants in the Qassim province of Saudi Arabia. The dataset was gathered by a Microsoft form survey from different customers. They used the KNN, RF, SVM, LR, and NB algorithms to classify the sentiment of reviews. The outcomes revealed that the SVM algorithm has the best accuracy with 89.0% among other algorithms.

Furthermore, the author in [13] built a Discriminative Multinomial Naïve Bayes (DMNB) model for classifying Arabic tweets into two polarities, which are positive and negative. They used a public Twitter Arabic dataset. The experiment was evaluated by applying a 5-fold cross-validation technique. The author compared the DMNB model with the other machine algorithms which were applied in related work and used the same dataset. The outcomes showed that the DMNB model has good accuracy, with 87.2%. The authors in [14] studied the SVM model for Arabic sentiment classification into two polarities, which are negative and positive. The dataset was collected from Twitter, which covers multiple social issues in Saudi Arabia. The results revealed that the SVM algorithm achieved satisfactory performance, with an accuracy of 89.83%. Furthermore, in [15], the researchers presented a model to study depressive emotions by using a dataset gathered from Twitter in the Gulf region. This study was based on the Patient Health Questionnaire (PHQ-9) and CES-D to diagnose the depression sign of Arabic tweeters. They evaluated the performance of the RF, AdaBoostM1, Liblinear, and NB algorithms to classify the dataset into two polarities: depressed and non-depressed. The Liblinear algorithm has the best accuracy, at 87.5%, among other algorithms.

The researchers in [16] built a model that combined the lexicon-based and machine learning methods for Arabic tweets sentiment classification. The SVM, RNN, and LR algorithms were used for tweet classification. The model was tested on a variety of datasets, including the Arabic Sentiment Tweets Dataset (ASTD), Mini Arabic Sentiment Tweets Dataset (MASTD), An Arabic Speech Act and Sentiment (ArSAS), Arabic Gold Standard Twitter Data for Sentiment Analysis (GS), Syrian Tweets Corpus, and the Twitter dataset for Arabic Sentiment Analysis (ArTwitter). They found that the RNN algorithm achieved the best accuracy of 73.67% in 3-class classification datasets, while the LR achieved the best accuracy of 83.73% in 2-class classification datasets.

Moreover, in [17], the authors proposed an improved method for Aspect-Based Sentiment Analysis (ABSA) of Arabic hotel reviews. They used supervised machine learning algorithms, comprising the KNN, SVM, NB, DT, and Bayes Networks. In this paper, the

evaluation findings revealed that all the proposed approach's classifiers surpassed the baseline approach. Moreover, the SVM algorithm attained the best accuracy, which is 95.4%, compared to other algorithms.

The authors in [18] studied Facebook comments written in MSA and Moroccan dialectal. They evaluated and contrasted two Arabic SA strategies. The first strategy is the traditional approach, which recognizes all Arabic texts as homogeneous. However, the second strategy required classifying the text prior to sentiment classification, which is based on language forms: standard and dialectal Arabic. Furthermore, the SVM and NB machine learning algorithms were applied in this study. The experiment obtained that the NB algorithm has better accuracy than SVM with 84.56% for MSA sentiment classification.

2.2. Arabic Sentiment Analysis Using the Deep Learning Approaches

The deep learning technique recently gained popularity in many areas, and researchers and data scientists have increasingly used the deep learning technique to address the issue of sentiment analysis. In addition, deep learning algorithms have been shown efficiency and productivity with massive datasets [10].

The authors in this research [10], proposed a model of deep learning comprised of LSTM, GRU, and ensemble techniques to study Arabic sentiment. They divided the model into three main sections. In the first section, the text documents are represented by the pre-trained word embedding technique to real-valued vectors. Then, they fed the vectors into LSTM and GRU networks to discover the high-level representations. In the last section, the result from LSTM and GRU networks will be input to the voting-based ensemble technique for classifying the Arabic sentiment. Furthermore, six available datasets have been used to assess the performance of the model. The outcomes of this model scored 94.32% accuracy, which is the best performance in the context of the outcomes of the state of the art of all six datasets that were selected.

In addition, in [19], the authors applied sentiment analysis to Moroccan datasets by using deep learning and traditional machine learning algorithms. The deep learning techniques were CNN and LSTM, while the traditional machine learning techniques were SVM, Maximum Entropy, and NB. The experiment revealed that the performance of deep learning algorithms is superior to the traditional machine learning algorithms. The findings demonstrated that the CNN algorithm obtained the highest accuracy of 99% with normalization and removal of stop words techniques.

In [20], the authors evaluated the LSTM deep learning algorithm on the Arabic Tweet dataset from the University of California, Irvine (UCI) Machine Learning Repository database. The LSTM was utilized to classify the sentiment of Arabic tweets into two polarities (positive and negative). The experiment demonstrated that the LSTM model outperformed the previous studies which used the same dataset by traditional machine learning algorithms with a score of 89.8% in terms of accuracy.

Moreover, the authors in [21], suggested a model of deep learning based on one layer of the CNN algorithm and two layers of the LSTM algorithm for classifying Arabic tweets sentiment. Furthermore, the FastText (Skip-gram) word embedding was considered as an input layer into the model to transform the words into vectors. To enhance the result, the model was passed to the SVM technique to produce the classification result. The experiment showed that the proposed model achieved a good result with 90.75% accuracy.

The researchers in [22], presented a model comprised of the CNN and LSTM algorithms for Arabic sentiment analysis. They applied the experiment to five existing datasets. In addition, the word2vec embedding was used as an embedding layer for the model. The result revealed that three datasets out of five, which are Arabic Health Services (Sub-AHS), Arabic sentiment tweets dataset (ASTD), and Opinion Corpus for Lebanese Arabic Reviews (OCLAR), achieved the best accuracy with 96.8%, 79.18%, and 90.3%, respectively, compared with related works that used the same datasets.

In another study, the authors in [23], applied a deep learning algorithm, which is the Simple Recurrent Unit (SRU), with the attention mechanism on the Large Scale Arabic

Book Reviews (LABR) dataset for sentiment classification. They evaluated their model by comparing it with the GRU algorithm. The result was that the proposed method surpassed the GRU algorithm and all the state-of-the-art algorithms with 94.53% in terms of accuracy.

The authors in [24] analyzed the emotions of Arabic tweets by applying deep learning methods. The SemiEval dataset is utilized in this experiment. The authors proposed categorizing the statements based on four emotions, including happiness, sadness, fear, and anger. They made a comparison between the deep learning algorithm, which is the CNN algorithm, and machine learning algorithms, the Multi-Layer Perceptron (MLP), SVM, and NB algorithms, to classify the emotions of Arabic tweets. The result revealed that the CNN algorithm outperformed the machine learning algorithms by 99.82% in terms of accuracy.

Researchers in [25] applied a deep neural network model consisting of a combination of a bi-directional LSTM algorithm and a CNN algorithm for Arabic dialectal sentiment analysis. Three datasets were used in this model, including ASTD, Shami-Senti, and LABR. The results demonstrated that the proposed model outperformed the baseline models. The proposed model achieved 93.5% accuracy on the Shami-Senti dataset.

In [26], the authors used deep learning to study Arabic sentiment analysis. They presented an Arabic labelled corpus for the tweets that are written in MSA and Egyptian dialects. The corpus is annotated manually into two polarities, which are negative and positive, to study the sentiment analysis. In the proposed corpus, the CNN, LSTM, and Region-based Convolutional Neural Network (RCNN) models are applied for classifying the sentiment of tweets. The outcomes showed that the LSTM technique has better accuracy than other algorithms, with a score of 81.31%.

The researchers in [27] applied CNN deep learning algorithm and three machine learning algorithms, the NB, SVM, and LR algorithms, to a big dataset to study the Arabic sentiment analysis. The dataset has been created from 13 Arabic datasets available specialized to sentiment analysis from various domains. Furthermore, the experiment revealed that the CNN algorithm outperforms the machine learning techniques by 94.33% in terms of accuracy in classifying Arabic sentiment.

In another study, the authors in [28] proposed the SEDAT (Sentiment and Emotion Detection in Arabic Text) model. The sentiment and emotional strength were predicted in Arabic tweets by using the CNN-LSTM model. They used the SemEval-2018 dataset to detect the emotion's strength and sentiment. The SEDAT model obtained better performance than the baseline model.

The authors in [29] applied the CNN, LSTM, and ensemble deep learning algorithms for Arabic tweets sentiment analysis. The AraVec (word embedding) was used on the ASTD dataset. The results revealed that the ensemble deep learning algorithms achieved the highest F1 score of 64.46% when compared to the deep learning algorithms used in previous works with the same dataset.

In [30], the authors examined the advantage of combining the CNN and LSTM deep learning models to analyze Arabic sentiment on four datasets. They studied many sentiment analysis levels including character level, character n-gram level, and word level. In this study, the character n-gram level, and word level achieved satisfactory results of sentiment classification. The outcomes showed that the model performance improved in the Main-AHS and Sub-AHS datasets with 94.24% and 95.68% in terms of accuracy, respectively, compared with other previous works carried out on the same datasets.

2.3. Sentiment Analysis Related to COVID-19

These days, COVID-19 is one of the hottest topics in the world. Therefore, researchers have decided to publish their research in this field. In the latest studies of sentiment analysis related to COVID-19, the author in [31] focused on Arab women to study their depression symptoms during COVID-19 by using a deep learning algorithm, which is the RNN algorithm. The author studied the tweets of Arab women to forecast depression from

their tweets. The results showed that the RNN algorithm has the best accuracy of 72% among the state-of-the-art models.

The authors in [32] studied sentiment analysis to identify and analyze people's emotions relating to COVID-19 by using deep learning and machine learning approaches to Arabic tweets. The researchers used feature extraction, involving word embedding and TFIDF methods, before applying the classifiers to the data because the machine learning and deep learning approaches do not handle the text in its normal form, so it must be converted to numeric form by using one of the feature extraction methods. Furthermore, the LSTM deep learning algorithm and the NB machine learning algorithm were applied to classify the sentiment. The experiment results showed that the LSTM achieved better results with 98.9% accuracy.

In [33], the authors performed classification of COVID-19 tweets' sentiments by applying a deep learning LSTM classifier and five supervised machine learning classifiers: RF, SVM, eXtreme Gradient Boosting (XGBoost), DT and Extra Tree Classifier (ETC). TextBlob Python library was applied to find the sentiments of tweets. Moreover, the authors used TF-IDF, Clove and Bag of Word (BOW) techniques as feature extraction and suggested a feature extraction technique by combining BOW and TF-IDF features. The authors found that the ETC algorithm accomplished the highest accuracy, which was 93% with the suggested feature extraction technique.

The authors in [34] studied the Arabic sentiment analysis to predict popular awareness of the prevention measures in five regions in Saudi Arabia to reduce the prevalence of COVID-19. The Arabic tweets were collected from multiple regions, the north, south, west, east, and middle regions, in Saudi Arabia by using the Twint tool during the period from 23 March 2020 until 21 June 2020. Furthermore, the authors used machine learning algorithms which are KNN, NB and SVM with feature extraction techniques for classifying the sentiment of Arabic tweets. They found that the middle region had the lowest level of awareness of the prevention measures for COVID-19, while the south region had the highest level. Moreover, the results revealed that the SVM algorithm with the TF-IDF feature extraction technique obtained 85% in terms of accuracy.

The authors in [35] suggested a novel NLP with an LSTM model for analyzing the semantic topic and sentiment comments about COVID-19 on the Reddit platform. This analysis was confined to the text in English. Furthermore, the classification of comments related to COVID-19 was divided into very positive, positive, very negative, negative, and neutral sentiments. They assessed the proposed LSTM method with traditional machine learning algorithms: the SVM, LR, Naive Bayes, and KNN algorithms. The results demonstrated that the proposed LSTM method accomplished the most, with 81.15% accuracy.

The researchers in [36] constructed an LSTM deep learning model for sentiment and emotion analysis classification on a Twitter dataset related to COVID-19. The first level in this model targeted classifying tweets into negative and positive. Then, the labelled tweets were utilized as an input into the next level to figure out the specific emotions such as joy, surprise, sadness, fear, anger, or disgust. The experiment's result showed that the LSTM with the word embedding model scored 76% in terms of accuracy.

In addition, in [37], the author applied an SVM classifier, which is a machine learning algorithm, a Stacked Gated Recurrent Units (SGRUs) algorithm, a Stacked Bi-GRU (SBGRU) algorithm, an AraBERT algorithm, and the Ensemble deep learning algorithm, which are deep learning algorithms, in order to classify the sentiment of Arabic tweets. Ten million Arabic tweets related to the COVID-19 pandemic were gathered in April 2020. The experiment outcomes indicated that the ensemble deep learning algorithm surpassed other algorithms by 90% in terms of accuracy. The authors in [38], used the NB and LR machine learning classifiers for classifying the public sentiment of COVID-19 English tweets. The authors collected tweets from February to March 2020, and the outcomes of the analysis showed that the fear of the pandemic increased significantly. On the other hand, the NB outperformed the logistic regression score by 91% in terms of accuracy.

In [39], the authors utilized the tweets related to COVID-19 to examine and study the interests, feelings and discussions of the public toward the COVID-19. The tweets were collected from 7 March to 17 April 2020, by using 20 hashtags related to COVID-19. Moreover, the authors applied an unsupervised machine learning method which is Latent Dirichlet Allocation (LDA) to extract the topics and concerns related to COVID-19. They noticed that multiple subjects were continuously popular on Twitter, such as “negative psychological reactions” (e.g., anger and fear), “preventive measures” and “the confirmed cases and death rates”. Researchers in this study [40], used unsupervised learning methods, the K-means and Mini-Batch k-means clustering algorithms, to study the influence of social distance on the public in the time of COVID-19 by analysis of tweets sentiment. They revealed that the majority of tweets in the United States, Nigeria, Australia, England, and Canada were neutral, according to the sentiment study. On the other hand, most of the tweets from India and Italy were positive. Moreover, the authors compared the K-means and Mini-Batch k-means clustering algorithms and the outcomes showed that the k-means outperformed the mini-batch k-means, while Mini-Batch k-means saved a significant amount of time when constructing clusters.

Furthermore, in [41], the authors presented the LSTM and GRU deep learning algorithms for analyzing Arabic tweet sentiments. They used an available dataset related to COVID-19 tweets where the tweets were collected based on Saudi hashtags. The authors annotated the polarity of tweets by using the available lexicon named AraSenTi, which is a corpus of 2.2 million tweets. The Global Vectors (GloVe) word embedding was applied as a feature extraction in this study. The outcomes revealed that the GRU classifier had better results than the LSTM, with an F1 score of 81%.

In [42], the authors proposed a hybrid framework to analyze tweet sentiments surrounding COVID-19. They used multiple machine learning algorithms and LSTM deep learning algorithms. According to the findings of this study, the LSTM has the highest accuracy of 83% when compared to other machine learning algorithms. Further, the authors in [43] investigated the sentiment analysis of COVID-19 tweets using ensemble machine learning. The finding showed that the stacking algorithm achieved the best F1-score of 83.5%.

2.4. Differences and Similarities of the Study over Previous COVID-19 Sentiment Analysis

Following are the differences between this study and the previous studies on COVID-19 and sentiment analysis:

1. The proposed research focuses on Arabic tweets’ sentiments related to, during and prior to COVID-19 in Saudi Arabia.
2. The dataset was collected from three specific regions in Saudi Arabia (with Saudi dialect and MSA), which are Riyadh, Jeddah, and Dammam.
3. The Arabic tweets were analyzed for sentiment in two periods: before and during the COVID-19 pandemic in Saudi Arabia.
4. The BiLSTM deep learning algorithm was chosen to classify Arabic sentiment analysis and it had not been investigated earlier for the said problem.
5. Multiple word embedding techniques are employed and the performance of these techniques compared in the developed dataset.

The reviewed literature was carefully selected for the current study based on the following similarities:

1. The proposed research aims to study sentiment analysis related to the COVID-19 era, such as the related works in the section on sentiment analysis related to COVID-19.
2. Deep learning algorithms are carried out for Arabic sentiment analysis.
3. Twitter is the source of the dataset.

3. Methodology

In this study, the methodology utilized to conduct sentiment analysis on the Arabic datasets has five steps as presented in Figure 1. In the first step, the data were collected from Twitter. Subsequently, the datasets were annotated and cleaned. Then, word embedding was applied to convert the text into vectors. Afterwards, the deep learning algorithms CNN and BiLSTM were implemented to classify the tweets into three polarities: 1, 0, and -1 . Finally, the experiment's performance was evaluated by applying different metrics such as accuracy, F-score, recall, and precision. All the steps will be described in detail in the next sections.

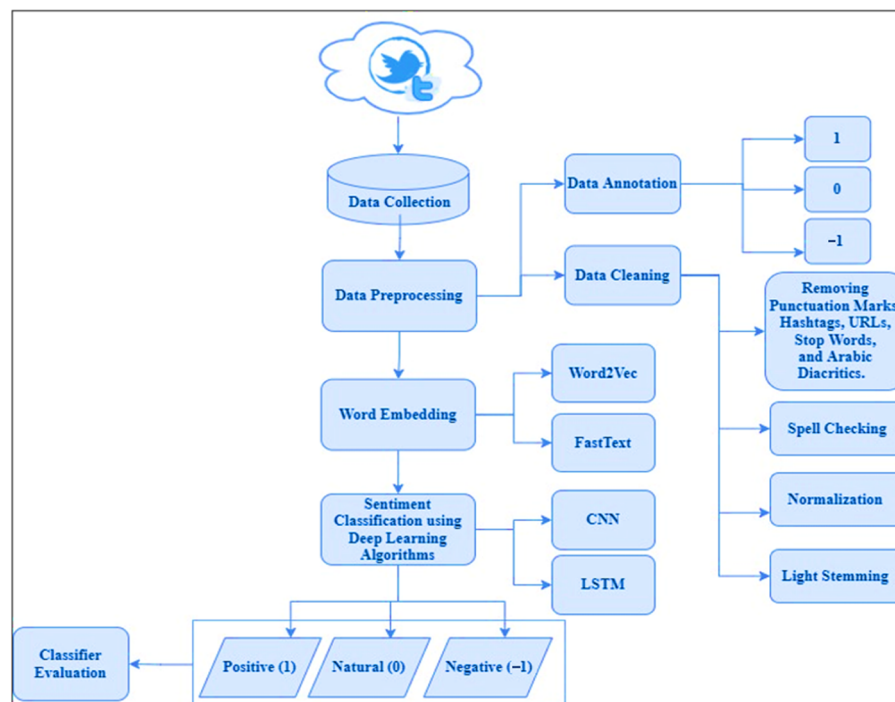


Figure 1. The proposed methodology of Arabic sentiment analysis.

3.1. Data Collection

For this research work, the dataset collected related to a large event in the world known as “Coronavirus (COVID-19)”. The SARS-CoV-2 virus is the cause of the COVID-19 disease. The dataset was obtained in the form of Arabic tweets from the Twitter platform. Twitter is one of the widely used social media sites in Saudi Arabia [44]. The data were collected by Twint. Twint is an Open Source Intelligence (OSINT) tool [45]. It is written in Python and is utilized to gather tweets without using Twitter’s API while avoiding most API limitations. Furthermore, the tweets were collected in two periods. In the first period, the tweets were collected before the COVID-19 pandemic in Saudi Arabia from 1st October 2019, until 28th February 2020. In the second period, tweets were collected at the beginning of the epidemic in Saudi Arabia, from 3rd March 2020, until 3rd March 2021. The aim of collecting the tweets in two different periods is to investigate the effect of COVID-19 in Saudi Arabia by comparing the sentiment analysis before and during the pandemic. Additionally, the dataset in this study includes 157,214 tweets and corresponding replies. During the COVID-19 pandemic period, tweets were collected using specific keywords such as “كورونا السعودية” and “كورونا”, whereas, before the pandemic period, tweets were collected using general words such as happy, sad, like, hate, and feel (أكره, حزين, سعيد, أحب, أشعر). On the other hand, the latitude and longitude have been identified for the three largest cities in Saudi Arabia, which are Riyadh, Jeddah, and Dammam, during the

collection of the data to ensure that the dataset is within the range of Saudi Arabia. The Twint tool commands used to collect data within the range of Saudi Arabia include:

- For the Riyadh region, this command was used with the corona (كورونا) keyword: "twint-g = "24.63333,46.71667,50 km" -s كورونا --since "2020-3-2" -o file.csv -csv".
- For Dammam region this command was used with the corona (كورونا) keyword since the epidemic started: "twint-g = "26.39222,49.97778,50 km" -s كورونا --since "2020-3-2" -o dammam_corona_file.csv -csv".
- For the Jeddah region this command was used with the corona keywords since the epidemic started: "twint-g = "21.54472,39.17611,50 km" -s كورونا --since "2020-3-2" -o jeddah_corona._file.csv -csv".

For other keywords or dates, we used the same commands, but we have changed the keyword or the date according to the situation.

3.2. Data Preprocessing

3.2.1. Data Annotation


A process of labelling text based on its context into three classes, which are positive, negative, or neutral; or two classes, which are positive or negative. In this study, four annotators manually annotated the tweets. Each tweet may be classified as positive (1), negative (-1), or neutral (0). The meeting was held with the annotators, and they were presented with the guidelines for labelling tweets. The guidelines are outlined below, along with the explanation for each one:

1. The author's viewpoint, not the annotator's, should be considered when annotating the tweets. In addition, the label (positive, negative, or neutral) should be chosen based on the tweet's context.
2. The news should be labelled as neutral even if it contains positive or negative information because news is not considered subjective.
3. If it is not clear to which label the tweet belongs, do not guess; instead, send it to the chat group and discuss it with other annotators, then choose the label with the most votes.
4. Because the mentions and hashtag signs were deleted, the subject of the tweet may not always be evident, but the sentiment could still be identified.
5. Delete tweets that contain advertisements or are incomplete due to download issues or other factors.
6. Delete tweets that contained non-Saudi dialects because the focus of this study was on sentiment analysis of Saudi dialects and MSA dialect tweets.
7. Correcting spelling errors in tweets that the annotator may encounter due to spelling errors or author abbreviations. Tables 1 and 2 present examples of annotated tweets for all datasets.

Table 1. Examples of annotated tweets in the period before COVID-19.

Tweets	Sentiment
صباح جديد ويوم خفيف سعيد مليون انجاز بإذن الله ☺☺☺	1
احس قلبي انقبض حسبي الله	-1
مطعم صيني ياباني اسمه لوتس في فندق الهيلتون ☹️ أحد جربه؟ شفت المنبر ومشييت لأنني لحالي وما أحب اكل الا مع قروب خصوصا إذا مطاعم زي كذا	0

Table 2. Examples of annotated tweets in period during COVID-19.

Tweets	Sentiment
الله يعز المملكة  الحمد لله اخذنا اول جرعة من لقاح كورونا	1
الاكتئاب قاعد ينتشر اكثر من كورونا.	-1
س: هل أخذ الزنك يقيني من فيروس كورونا ج: لا س هل اذا أخذت الكلوروكوين او هيدروكسيكلوروكوين بحميني من الكورونا ج لا البرتوكول ليس للاستخدام الشخصي بل للمختصين في المستشفيات ولا تحاول اقتناء اي من الأدوية والأدوية لا تغني عن العناية الصحية والعزل وهي الأساس في اغلب الحالات	0

3.2.2. Data Cleaning

Data preprocessing or data cleaning is considered a major stage during data analysis because it is important to eliminate any unwanted, noisy, or unnecessary data in order to achieve good sentiment classification. The steps that were used to clean the data in this study are:

1. Removing the Arabic diacritics, Table 3 shows the diacritic marks in the Arabic language.

Table 3. Arabic diacritic marks.

Diacritic Marks	Characters
Fatha	َ
Tashdeed	ّ
Tanwin Fath	ً
Damma	ُ
Tanwin Damm	ٌ
Kasra	ِ
Tanwin Kasr	ٍ
Sukun	◌

2. Removing hashtags, user mentions, and URLs.
3. Removing punctuation marks, such as full stops and commas, is ineffective in detecting polarity.
4. Removing repeated tweets because there is no need to classify repeated tweets; it is enough to classify one.
5. Removing non-Arabic and commercial tweets.
6. Removing the elongation and repeating letters such as the word “مهممم” to become “مهم”.
7. After removing the repeated letters, some words may be affected, especially if the words have double letters in the original, such as “الله” becoming after removing the double letters “اله”, spell checking is important in these situations. In addition, the mistakes or abbreviations by users during writing need to be corrected.
8. Tokenization is one of the NLP tasks that divides a string of words into semantically relevant tokens by whitespace or punctuation characters.

9. Normalization is the method of reducing letters to their most basic form. The Arabic language is rich morphologically, so it needs normalization. The normalization form for some Arabic letters is presented in Table 4.

Table 4. Examples of some letters in normalization form.

Letter	Normalized Form
ا، آ، أ، إ	ا
ى	ي
ئ	ء
ؤ	ء
ة	ه
ك	ك

10. Stop word removal: Stop words are employed to structure language but do not contribute to its content. Some of the Arabic stop words, such as *من، هذا، الذي*.
11. Light stemming is a way of removing suffixes and prefixes from words without shrinking them to their root, such as the word “مرفوض” after applying light stemming will be “رفض”. Compared to the root stemming approach, light stemming was revealed to have good outcomes with text classification problems [46]. Furthermore, light stemming has been shown to be effective in several studies [47–49]. However, the light stemmer algorithm proposed by Motaz Saad [50], was applied in this study. The number of tweets before and after preprocessing is summarized in Table 5. In addition, the examples of tweets before and after preprocessing are presented in Table 6.

Table 5. The size of the collected tweets before and after the cleaning process.

Files	Before Cleaning		After Cleaning	
	Before COVID-19	During COVID-19	Before COVID-19	During COVID-19
Riyadh File	23,307	44,073	13,487	25,860
Jeddah File	17,788	51,386	13,244	20,912
Dammam File	7521	13,139	6887	9613

Table 6. Examples for tweets before and after processing.

Tweets before Processing	Tweets after Processing
قررت ما اشارك فيها مب عن نحاسة بالعكس انا اكثر انسانة كنت احب اوري كيف مبسوطه بهاللحظة الاكثاب قاعد ينتشر اكثر من كورونا.	قرت اشارك مب نحاس عكس انسان احب اور مبسوط بهالحظ اكتباب قاعد ينتشر كورونا

3.3. Word Embedding

Word embedding is a language modelling method that uses vectors to represent words or sentences. In other words, it is a form of text representation in which words with the same meaning are represented similarly. The goal of word embedding is to recognize a word’s context in a document, its semantic and syntactic similarities, and its relationship to other words. Word embedding is one of the significant elements for addressing some tasks for sentiment analysis and other NLP tasks. Word embedding is considered the first layer for data processing in deep learning methods because the deep learning algorithms do not deal with the text directly and must be represented as vectors by using one of the word embedding techniques [51,52]. Furthermore, recent research has shown that

embedding techniques give better quality outcomes in all NLP applications than classical techniques of feature extraction [53]. Word embedding has multiple types that are used in NLP applications such as Word to Vector (word2vec), FastText, and Global Vector (GloVe). The word2vec and FastText models were applied in this study; they are among the most common word embedding models

3.3.1. Word2vec

Word2vec is one of the widely used techniques in NLP fields for learning word embedding by applying shallow neural network techniques on a huge corpus of text. It was unveiled by Google in 2013 by Tomas [53]. It is used to process and transform text data into vectors to be applied as features in deep learning algorithms. Word2vec comes in one of two forms: Common Bag of Words (CBOW) or skip-grams techniques. In the architecture of the CBOW technique, the CBOW forecasts the present word from the neighboring context words window. On the other hand, the skip-gram technique forecasts the surrounding context words window using the current word. Skip-gram is good at representing infrequent words and works effectively with a limited amount of data. Furthermore, the CBOW technique works more rapidly than the skip-gram technique in training and provides better representations of words that are used more frequently [53]. The architecture of skip-gram and CBOW techniques is illustrated in Figure 2.

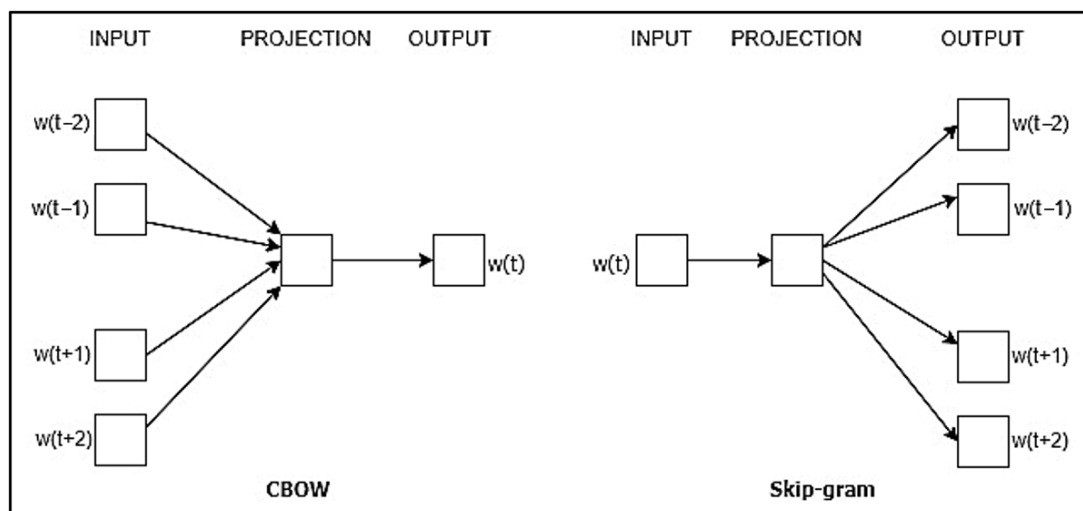


Figure 2. The architecture of CBOW and SG models.

3.3.2. FastText

The FastText model was introduced by Facebook's AI research in 2016 [54]. The FastText model is an extension of the word2vec model, but is not the same. FastText divides words into n-grams (sub-words) instead of providing single words to the neural network. It treats a word as a collection of n-grams characters, with n ranging from one to the length of a word. For example, the 3-g for the word orange are represented as: "or, ora, ran, ang, ge". The total of all these n-grams will be the word embedding vectors for orange. Therefore, FastText takes into account the internal structure of words when learning word representation, which is useful in languages with complex morphology, such as Arabic [55]. One of the benefits of utilizing FastText is that it provides good word embedding for rare words and even words that are not observed during training. On the other hand, FastText has been frequently applied for a variety of classification applications, such as sentiment analysis [55]. Figure 3 shows the components of the FastText word embedding technique.

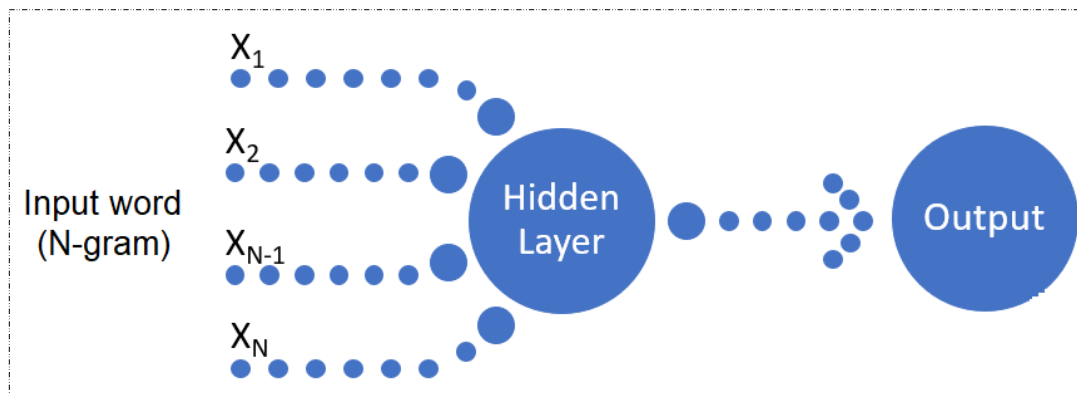


Figure 3. FastText architecture.

3.4. Description of the Proposed Deep Learning Techniques

3.4.1. Convolutional Neural Network (CNN)

Kim proposed the CNN algorithm for text classification in 2014 with a simple and powerful design [56]. A CNN is a form of deep learning algorithm and a feed-forward artificial neural network (where nodes’ connections do not form a cycle). Due to CNN’s capacity to capture the semantic and syntactic characteristics of a specific function, it has demonstrated outstanding performance in a variety of NLP-related activities, such as sentiment analysis [56] and [19]. CNN uses the input layer convolutional processes to extract features automatically, and weight sharing across all neurons improves the CNN’s learning ability [57]. CNN uses a variety of multilayer perceptron designed to require a minimal amount of preprocessing. Figure 4 illustrates the architecture of the CNN model for sentence classification proposed by [56]. The CNN model typically comprises four layers: an input layer, a convolutional layer, a pooling layer, and a fully connected layer. The convolutional layer usually comprises different convolution kernels to extract features from the input layer. To minimize computational complexity, the CNN model utilizes a pooling layer. The pooling mechanisms used by CNN minimize the stack output size from one layer to the next while preserving crucial information. Max-pooling is one of the most commonly used pooling techniques where the pooling window contains an element of max value. The outcome of the pooling layer is fed into the flattened layer, which integrates all features into a unified matrix. The flattened layer then maps the output to the following layers. The fully connected layer is mostly the last layer in the CNN model.

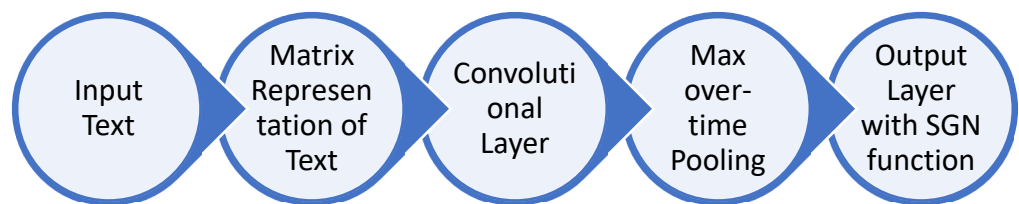


Figure 4. The architecture of the CNN algorithm with two channels for sentence classification.

3.4.2. Long Short-Term Memory Network (LSTM)

LSTM is a form of Recurrent Neural Network (RNN). Hochreiter and Schmidhuber introduced the LSTM in 1997 [58]. The LSTM algorithm is specifically designed to solve the problem of vanishing gradients that has been faced by the RNN algorithm. The principle of an LSTM operation is to hold the relevant data and discard the data that are not needed or beneficial for further prediction. Hence, it is characterized by remembering the previous values for any period [59]. LSTM has been demonstrated to be effective when used to solve sentiment analysis problems [60,61]. In the LSTM architecture, the hidden layer consists of memory blocks that are self-linked sub networks made up of many internet cells. By using

multi-gates, the cells were able to save and access information for an extended period [59]. The following are the gates that are inside each cell of the LSTM:

- Forget Gate:

The initial step in an LSTM network is a forget gate. The forget gate determines whether the information should be preserved or discarded from the cell state by applying the sigmoid function. The results of F_t are between 0 and 1. A result close to zero means discarding the information, while a result close to one means keeping the information. The forget gate F_t equation at the time (t) is as follows [62]:

$$F_t = \sigma(W_F X_t + U_F H_{t-1} + b_F) \quad (1)$$

where σ is the sigmoid function, X_t denotes the input at a time (t), W_F and U_F represent the weight matrix of forget gate, the bias is represented by b_F while the hidden activation described by H_{t-1} at a time ($t-1$).

- Input Gate:

The input gate defines the information that should be preserved within the cell state. Hence, the input gate has been used to update the cell state. The input gate I_t at the time (t) is computed as [62]:

$$I_t = \sigma(W_I X_t + U_I H_{t-1} + b_I) \quad (2)$$

where σ is sigmoid function, X_t denotes to the input at a time (t), W_I and U_I represents the weight matrix of forget gate, the bias is represented by b_I while the hidden activation described by H_{t-1} at time ($t-1$).

- Output Gate:

The output gate identified the output values that should be in the next hidden state. The equation for computing the output gate O_t is [62]:

$$O_t = \sigma(W_O X_t + U_O H_{t-1} + b_O) \quad (3)$$

where σ is sigmoid function, X_t denotes the input at a time (t), W_O and U_O represents the weight matrix of forget gate, the bias is represented by b_O while the hidden activation is described by H_{t-1} at time ($t-1$).

Moreover, the cell state C_t at the time (t) and hidden activation H_t at time (t) can be computed as [62]:

$$C_t = F_t \times C_{t-1} + I_t \times \tanh(W_C X_t + U_C H_{t-1} + b_C) \quad (4)$$

where \tanh is the hyperbolic tangent function.

Figure 5 presents the architecture of the LSTM model. On the other hand, LSTM models may come in a variety of forms. There are two main LSTM forms:

- Forward LSTM
- Bi-directional LSTM

The forward LSTM classifiers are unidirectional LSTMs that manipulate the inputs in only one direction from past to future, whereas the bi-directional LSTM manipulates the inputs in both directions, forward and backward, to persist the information [63].

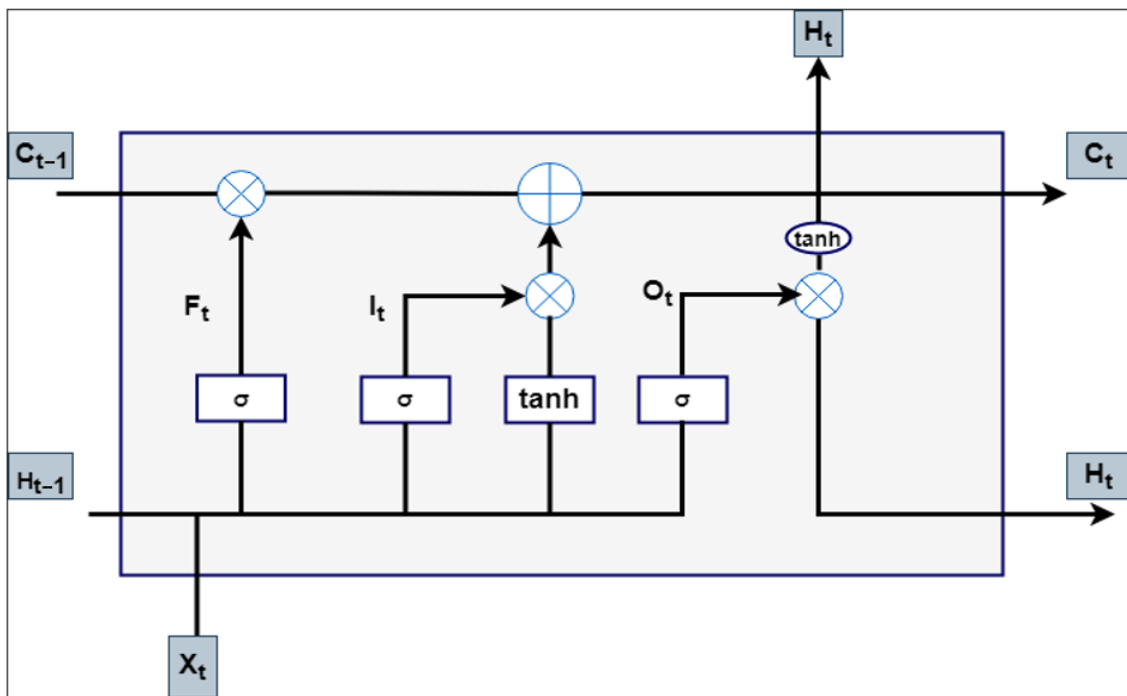


Figure 5. The architecture of LSTM cell.

Bi-Directional Long Short-Term Memory (BiLSTM)

BiLSTM is a technique that allows any neural network to store a sequence of inputs in both directions, backwards and forwards; thus, bi-LSTM differs from a standard LSTM in that inputs can flow in both directions. The output layer obtains information from both directions; the earlier sequences (backwards) and the following sequences (forward) at the same time. Therefore, it is highly beneficial when the input context is required, such as when the positive term follows a negation term [64]. Figure 6 shows the structural design of the Bi-LSTM model.

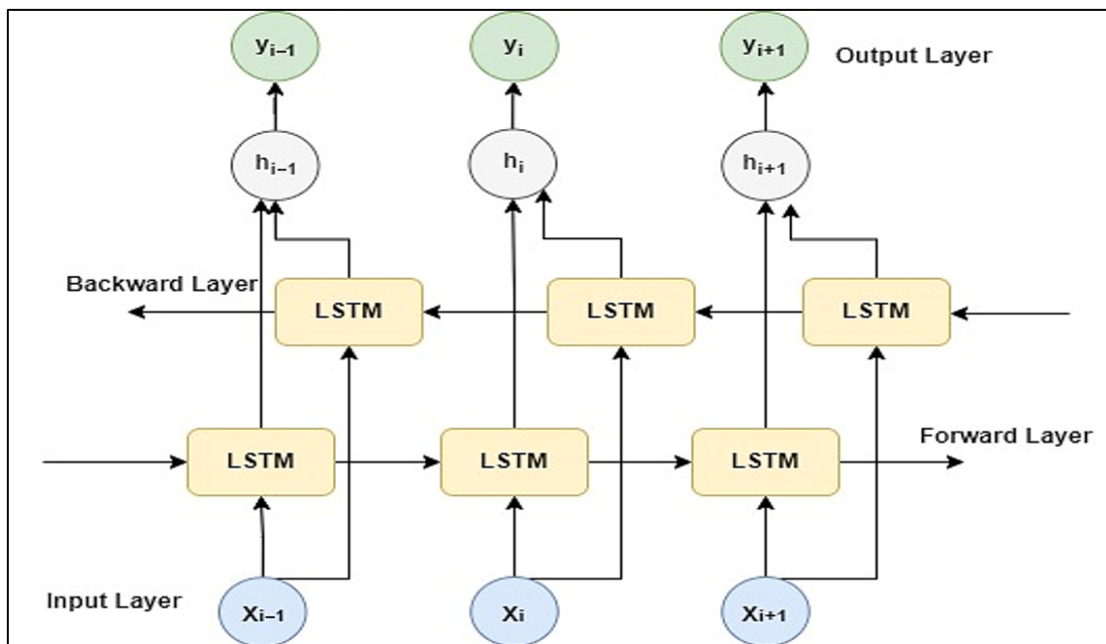


Figure 6. The Bi-LSTM model architecture.

3.5. Evaluation Metrics

To assess the performance of our model on the Saudi COVID-19 dataset, four measures are used: accuracy, F-score, recall, and precision.

- Accuracy: The result of dividing the number of true classified outcomes by the whole of classified instances. The accuracy is computed by the equation [65]:

$$Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + TrueNegative + FalsePositive + FalseNegative} \quad (5)$$

- Recall: The percentage of positive tweets that are properly determined by the model in the dataset. The recall calculated by [65]:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \quad (6)$$

- Precision: The proportion of true positive tweets among all forecasted positive tweets. The equation of precision measure calculated by [65]:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \quad (7)$$

- F-score: A harmonic mean of precision and recall. The F-score measure equation is [65]:

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \quad (8)$$

4. Experiment Results and Discussion

We have implemented the proposed techniques in the Python language with TensorFlow, Keras, and Sklearn Python libraries. Furthermore, the experiments ran on Windows 11 Pro with a 12th Gen Intel(R) Core(TM) i9-12900KF 3.20 GHz processor and 32.0 GB RAM.

4.1. Word Embedding Experiments

In this study, we made a comparison between multiple Arabic open source projects' pre-trained word embeddings, the AraVec [66], ArWordVec [67], and FastText [68] by using BiLSTM and CNN algorithms. Moreover, the experiments have been implemented on each one of the eight datasets because an embedding model may work effectively on a particular dataset, but not likewise on the other datasets.

The AraVec has been trained by [66] on one of three Arabic text resources: Wikipedia Arabic articles, WWW public websites, and Tweets. In addition, each text resource contains two models: one constructed with the CBOW technique and the other with the SG technique. Moreover, the total number of tokens exceeded the 3,300,000,000 that were used to construct the AraVec model. The AraVec-Twitter model with n-gram CBOW and SG architectures has been used in this research.

The other Arabic word embedding is the ArWordVec model, which was trained by [67] on Arabic tweets with two models, SG and CBOW. The FastText model was also trained by [68] on Wikipedia and common crawl used the CBOW architecture. The results of the comparison between all pre-trained word embeddings for all datasets in two periods are shown in Tables 7 and 8.

Table 7. The accuracy metric of CNN and BiLSTM algorithms when using different Arabic pre-trained word embedding techniques for dataset before COVID-19.

Word Embedding Model	Hyper-Parameters			Algorithms	Datasets before COVID-19 Period			
	Architecture	Vector Size	Window Size		Riyadh Dataset	Jeddah Dataset	Dammam Dataset	All Datasets
ArWordVec	SG	300	3	CNN	92.18	87.27	80.86	88.99
		300	5		91.92	87.35	81.11	88.94
		500	3		92.24	87.23	81.20	89.14
		500	5		92.10	86.96	80.90	88.71
	CBOW	300	5		92.43	86.72	79.43	88.72
		500	3		92.34	87.03	80.55	89.05
500		5	92.06		86.72	80.34	89.01	
AraVec	SG	100	3		92.09	87.70	80.60	89.21
		300	3		92.80	88.35	83.39	89.69
	CBOW	100	3		92.06	86.66	76.80	88.47
		300	3		92.00	87.29	81.43	89.23
FastText	CBOW	300	-		92.24	87.62	79.95	88.76
ArWordVec	SG	300	3	BiLSTM	91.28	85.83	80.61	88.37
		300	5		91.76	86.02	80.78	88.04
		500	3		90.92	86.50	81.29	88.42
		500	5		91.38	85.70	81.08	88.09
	CBOW	300	5		91.03	86.22	81.29	88.04
		500	3		91.75	87.04	81.10	88.35
500		5	91.27		86.63	81.34	87.67	
AraVec	SG	100	3		91.19	85.00	81.70	87.68
		300	3		91.99	86.83	82.25	88.74
	CBOW	100	3		91.49	86.94	81.96	88.07
		300	3		91.88	87.41	83.14	88.56
FastText	CBOW	300	-		91.05	84.57	80.10	87.49

Table 8. The Accuracy metric of CNN and BiLSTM algorithms when using different Arabic pre-trained word embedding techniques for the dataset during COVID-19.

Word Embedding Model	Hyper-Parameters			Algorithms	Datasets during COVID-19 Period			
	Architecture	Vector Size	Window Size		Riyadh Dataset	Jeddah Dataset	Dammam Dataset	All Datasets
ArWordVec	SG	300	3	CNN	78.27	81.62	70.19	77.52
		300	5		78.19	81.38	70.27	77.84
		500	3		78.02	80.87	69.95	77.73
		500	5		77.83	81.35	70.82	77.17
	CBOW	300	5		78.12	80.99	69.85	77.61
		500	3		77.90	80.98	70.11	77.70
500		5	78.31		80.87	70.61	77.80	
AraVec	SG	100	3		78.71	81.00	71.78	77.84
		300	3		79.52	82.77	72.28	78.98
	CBOW	100	3		77.78	81.12	70.43	77.17
		300	3		78.25	81.64	70.84	77.69
FastText	CBOW	300	-		78.49	81.70	71.05	78.53

Table 8. Cont.

Word Embedding Model	Hyper-Parameters			Algorithms	Datasets during COVID-19 Period			
	Architecture	Vector Size	Window Size		Riyadh Dataset	Jeddah Dataset	Dammam Dataset	All Datasets
ArWordVec	SG	300-3	3	BiLSTM	76.49	79.54	67.75	75.58
		300-5	5		76.27	79.41	67.78	75.99
		500-3	3		76.32	79.64	68.54	76.39
		500-5	5		76.61	79.67	68.05	76.12
	CBOW	300-5	5		77.00	79.63	68.29	76.69
		500-3	3		77.22	80.08	68.45	76.69
		500-5	5		77.18	79.92	68.83	76.81
		100	3		76.36	79.84	67.82	75.81
AraVec	SG	300	3	77.78	80.75	69.77	77.19	
		100	3	77.43	79.88	68.08	76.11	
	CBOW	300	3	78.03	80.63	69.83	77.24	
		300	-	75.51	79.25	67.27	74.87	
FastText	CBOW	300	-					

4.1.1. Word Embedding Results of the Datasets before the COVID-19 Pandemic Period

From Table 7 above, for the CNN model, the AraVec-SG-300-3 model surpassed all other pre-trained word embedding models of the Riyadh dataset by 92.80% in terms of accuracy. In contrast, the AraVec-CBOW-100-3 model attained the lowest performance, with 76.80% accuracy with the Dammam dataset compared with the other pre-trained word embedding models. Figure 7 illustrates the comparison results for selected Arabic pre-trained word embedding models with the CNN algorithm and the datasets before the COVID-19 pandemic. Furthermore, for the BiLSTM model, the AraVec-SG-300-3 model scored the highest accuracy, with 91.99% for the Riyadh dataset, while the FastText word embedding has the worst performance, with 80.10% accuracy for the Dammam dataset. We noticed that the AraVec-SG-300-3 word embedding model has the best results with the two algorithms of the Riyadh dataset. Figure 8 illustrates the outcomes of the comparison for selected Arabic pre-trained word embedding models by applying the BiLSTM algorithm and the datasets before the COVID-19 pandemic.

4.1.2. Word Embedding Results of the Datasets during the COVID-19 Pandemic Period

From Table 8 above, for the CNN model, the AraVec-SG-300-3 model has the best results, with 82.77% accuracy for the Jeddah dataset compared with other pre-trained word embedding models. In addition, the AraVec-CBOW-300-5 model attained the lowest performance, with 69.85% accuracy for Dammam dataset compared to other pre-trained word embedding models. Figure 9 demonstrates the results of chosen Arabic pre-trained word embedding models with the CNN algorithm and the datasets during the COVID-19 pandemic. Furthermore, for the BiLSTM model, the AraVec-SG-300-3 model achieved the highest accuracy for Jeddah dataset, with 80.75% accuracy. The FastText word embedding model has the lowest performance, with an accuracy of 67.27% for the Dammam dataset. Further, the AraVec-SG-300-3 word embedding model has been noted to have the best performance with all algorithms compared to other word embedding models. The comparison results charts of all Arabic pre-trained word embedding models with the BiLSTM algorithm and the datasets during the COVID-19 pandemic are presented in Figure 10.

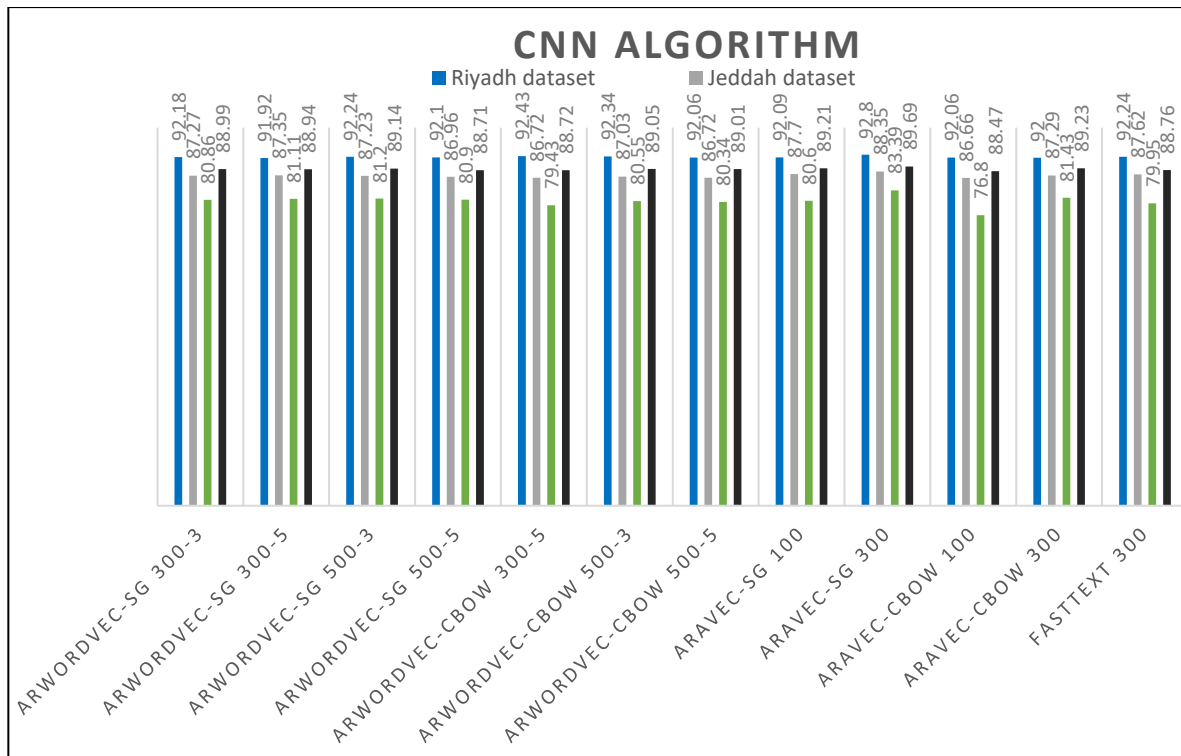


Figure 7. The Accuracies metric of CNN algorithm for comparing different Arabic pre-trained word embedding techniques for dataset before COVID-19.

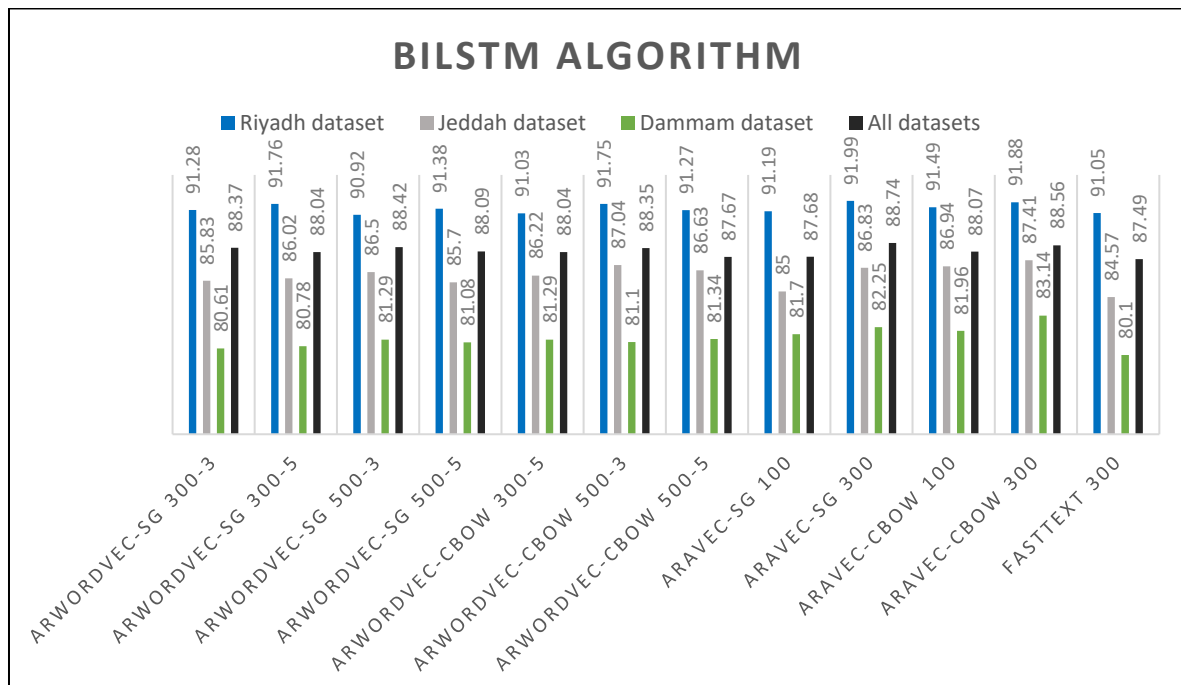


Figure 8. The Accuracies metric of BiLSTM algorithm for comparing different Arabic pre-trained word embedding techniques for dataset before COVID-19.

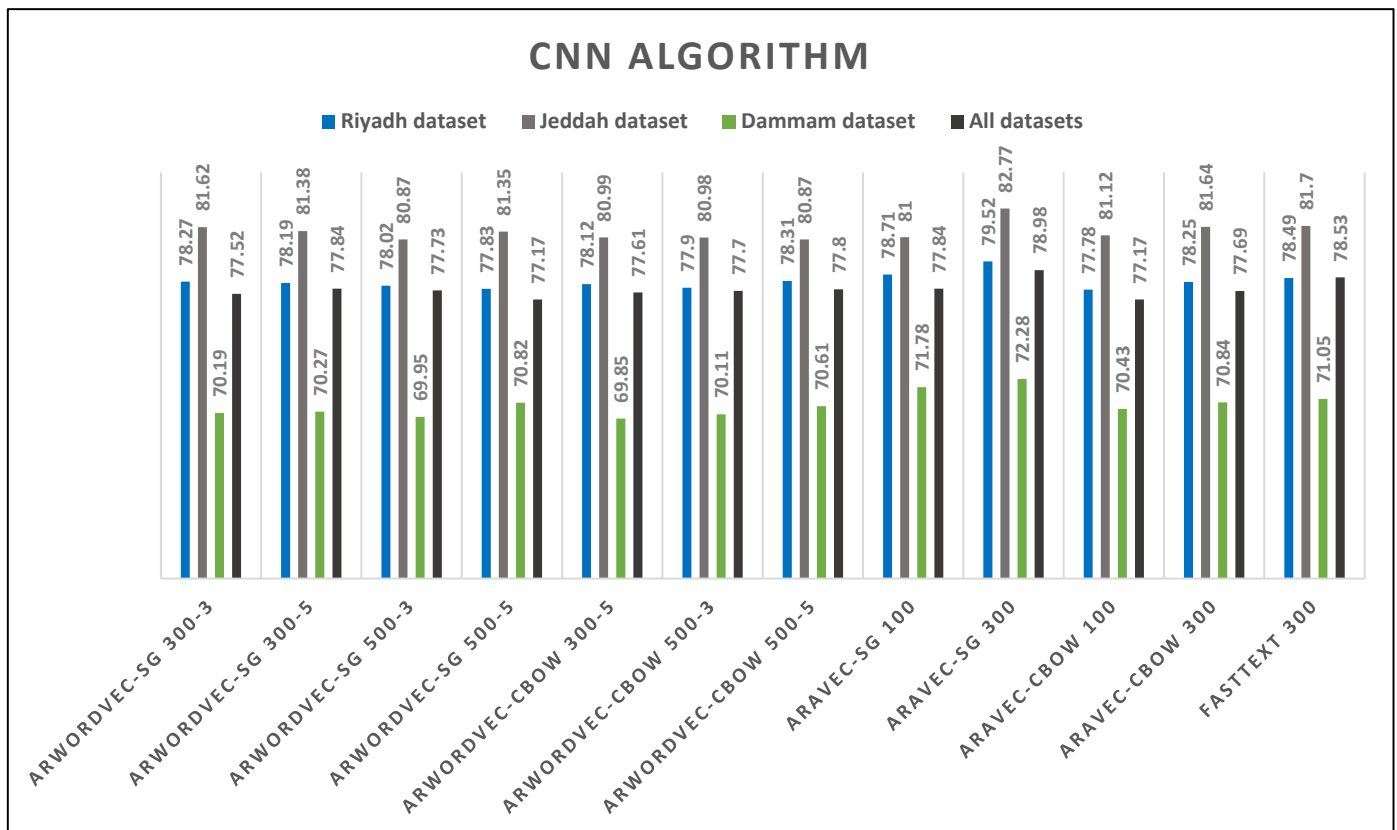


Figure 9. The Accuracies metric of CNN algorithm for comparing different Arabic pre-trained word embedding techniques for dataset during a COVID-19 pandemic.

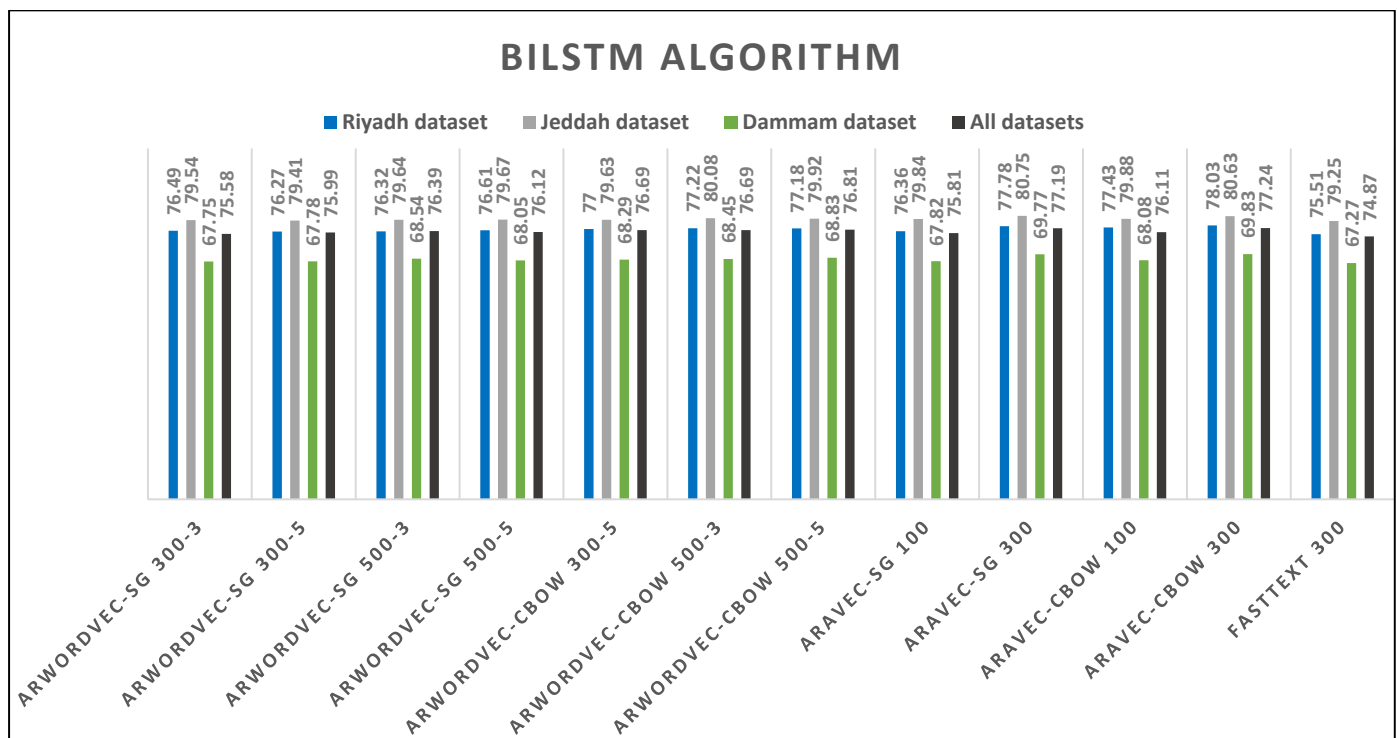


Figure 10. The Accuracies metric of BiLSTM algorithm for comparing different Arabic pre-trained word embedding techniques for the dataset during the COVID-19 pandemic.

4.2. Deep Learning Algorithms Results

After choosing the best hyper-parameters and the best Arabic word embedding models, in this section, the experiments were applied by using the deep learning algorithms CNN and BiLSTM on Arabic tweets for studying the impact of COVID-19 in Saudi Arabia. The datasets in this study feature two dialects: the Saudi dialect and the MSA dialect. The CNN and BiLSTM algorithms have been trained with different architectures. The selected architecture of the CNN model was applied in the experiments for all datasets. The architecture starts from the input layer, which is the word embedding layer, followed by a one-dimensional convolution layer. Then, one dimension of a global max pooling layer has been used to decrease the output dimensions. Following that, the dropout and dense layers were added, which flattened the output from all previous layers. Finally, the fully connected layer with a dense layer was added. In addition, the architecture of the BiLSTM model has been applied to all datasets in all experiments. The BiLSTM architecture begins as the architecture of the CNN model with the word embedding layer. Then, it is followed by the BiLSTM unit layer and dropout layer. The last layer is a fully connected layer with a dense layer.

The CNN and BiLSTM have been evaluated on the Arabic dataset by multiple matrices such as accuracy, precision, F-score, and recall. The findings showed that the CNN algorithm outperformed the BiLSTM algorithm in terms of accuracy, recall, precision, and F1-score across all datasets. Therefore, the CNN algorithm achieved the highest accuracy with 92.80% when evaluating the model with Riyadh dataset (before the COVID-19 period). On the other hand, the lowest accuracies achieved by CNN and BiLSTM algorithms were 72.28% and 69.83%, respectively, with the Dammam dataset (during the COVID-19 period). We also noticed that the performance of all models with Dammam datasets (during the COVID-19) period was the lowest in terms of all metrics compared with other datasets. Riyadh datasets in the two periods achieved the best results, with 92.8% and 91.99% accuracy compared to other datasets. Furthermore, the performance of datasets in the period before COVID-19 achieved better results than the performance of datasets during the period of COVID-19. Table 9 shows the results of all algorithms for all datasets in the two periods. In addition, Figures 11 and 12 present results charts of CNN and BiLSTM models for all datasets.

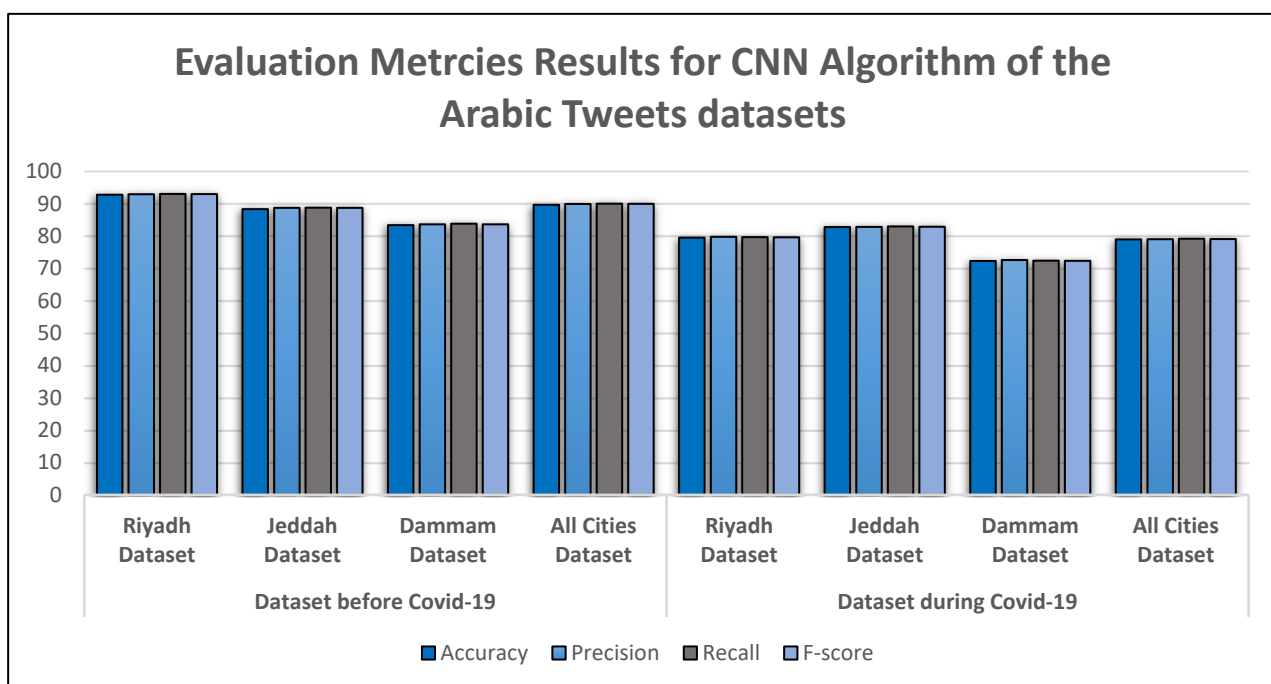


Figure 11. The Chart of Results of the CNN algorithm for all datasets.

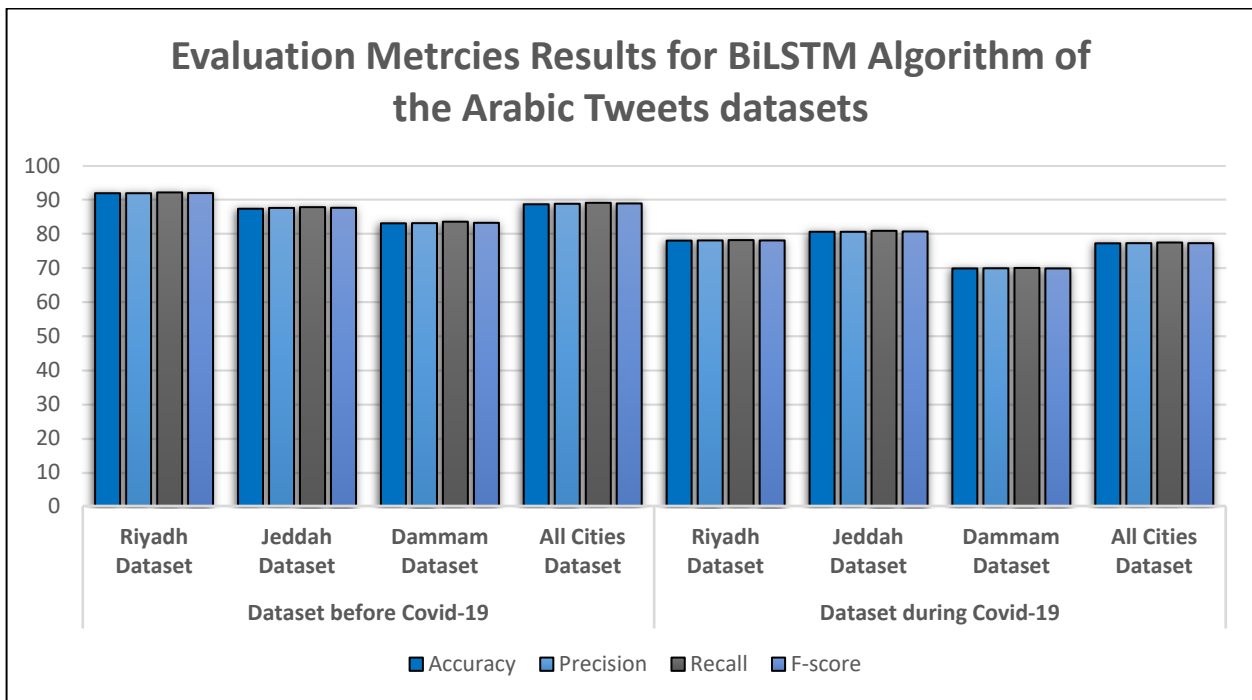


Figure 12. The Chart of Results of the BiLSTM algorithm for all datasets.

Table 9. The performance metrics of CNN and BiLSTM algorithms on datasets in all two periods.

Algorithms	Performance Metrics	Dataset before COVID-19				Dataset during COVID-19			
		Riyadh Dataset	Jeddah Dataset	Dammam Dataset	All Cities Dataset	Riyadh Dataset	Jeddah Dataset	Dammam Dataset	All Cities Dataset
CNN	Accuracy	92.80	88.35	83.39	89.69	79.52	82.77	72.28	78.98
	Precision	92.97	88.73	83.65	89.93	79.81	82.87	72.60	79.05
	Recall	93.06	88.77	83.81	90.03	79.66	83.01	72.39	79.18
	F-score	92.99	88.72	83.63	89.96	79.61	82.90	72.34	79.07
BiLSTM	Accuracy	91.99	87.41	83.14	88.74	78.03	80.63	69.83	77.24
	Precision	91.99	87.65	83.18	88.87	78.11	80.67	69.90	77.30
	Recall	92.25	87.88	83.63	89.16	78.20	80.93	69.99	77.48
	F-score	92.06	87.72	83.29	88.97	78.11	80.73	69.87	77.30

The confusion matrix and Receiver Operation Characteristic (ROC) curve are approaches used for evaluating classifier performance. Figure 13 depicts the confusion matrix of the CNN algorithm for the Riyadh (before COVID-19) dataset. In the confusion matrix, the predicted classes are shown in the rows while the true classes are shown in the columns. The blue cells indicate that the classifiers are working well, whereas the white cells indicate that they are not. Hence, the CNN classifier showed that it was performing well in the true classes.

In addition, Figure 14 presents the ROC curve of CNN for the Riyadh (before COVID-19) dataset. Superior performance is shown in the ROC curve if a greater area is under the ROC curve, and vice versa. Hence, the ROC curve of the neutral class (0) is better than the positive class curve (1) and negative class curve (2). Overall, the ROC curve for the CNN classifier denoted that the classifier performed well.

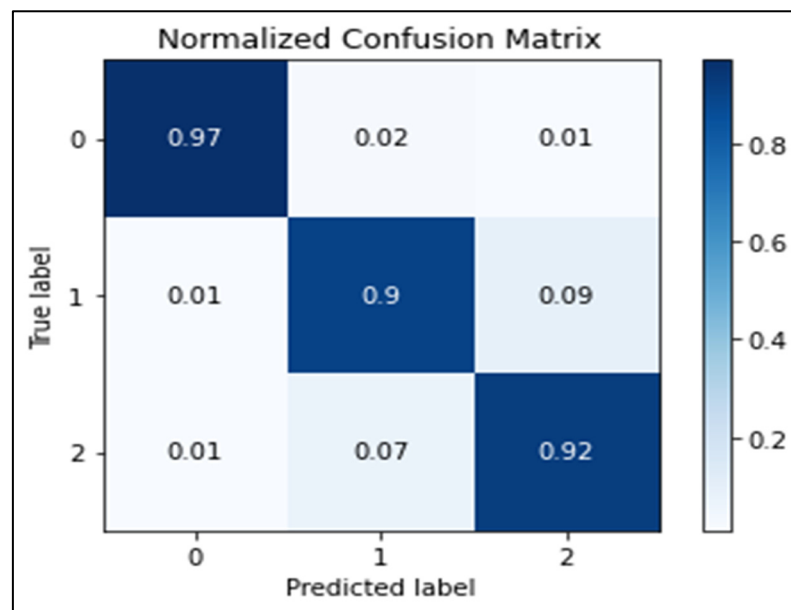


Figure 13. Confusion matrix of CNN algorithm for Riyadh dataset (before COVID-19) dataset.

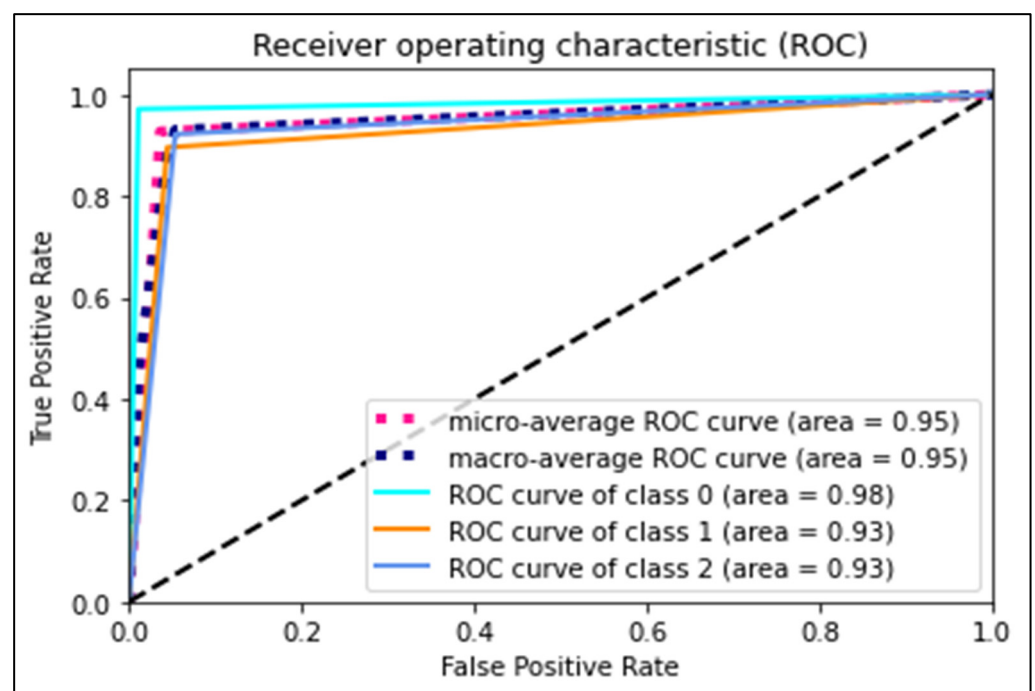


Figure 14. ROC curve of the CNN algorithm for Riyadh dataset (before COVID-19) dataset.

4.3. Word Cloud

In this research, a technique known as “word cloud” has been used to visualize and show the most frequent words in the text. A word cloud is a grouping of words that are shown in assorted sizes. It is a trendy way of illustrating the importance of words in a group of texts. The more commonly a word appears, the more space it takes up in the image. Furthermore, the word cloud is useful for gaining an understanding of what the collection of texts is about. Therefore, word clouds have been employed for visualizing the most frequently used words in our Arabic dataset. Figure 15 to Figure 16 below show the word clouds of all datasets before or during the COVID-19 pandemic. The common words of the dataset (before the COVID-19 period) were general words such as morning, people, my family, life, love, thanks, famous football teams (al-Hilal team), and a lot of

supplications that indicate that the Saudi people are religious by nature. Therefore, the visualization of the word cloud of the datasets (before the COVID-19 period) denoted that the Saudi people had a normal life throughout this period. According to the distribution of the datasets mentioned in Table 10, positive sentiments were the highest among other sentiments in all cities: Riyadh, Jeddah, and Dammam.

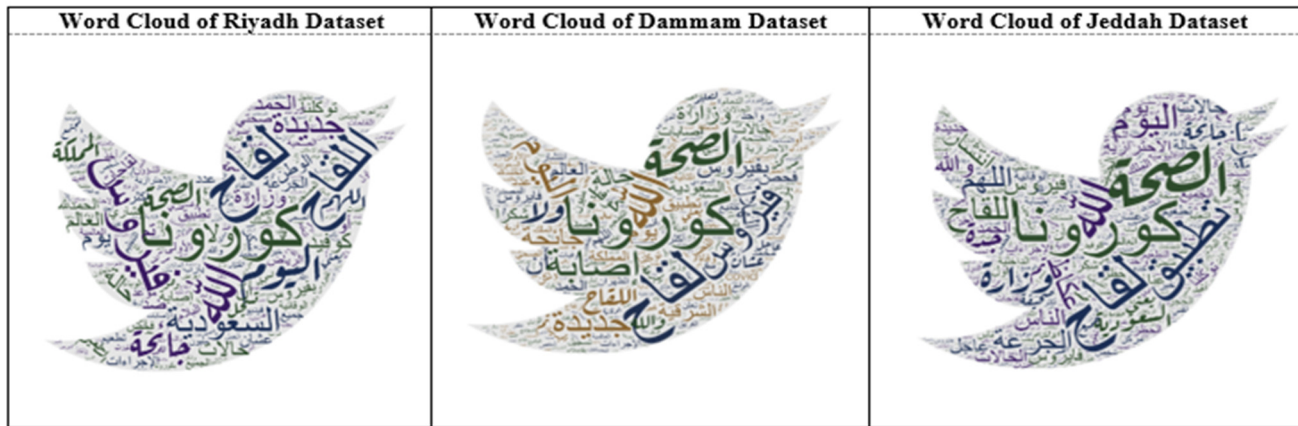


Figure 15. Word Clouds of datasets during COVID-19.



Figure 16. Word Clouds of datasets before COVID-19.

Table 10. Distribution of sentiment classes for (before COVID-19) dataset.

Dammam File (before COVID-19)		Jeddah File (before COVID-19)		Riyadh File (before COVID-19)	
Sentiment Class	Number of Tweets	Sentiment Class	Number of Tweets	Sentiment Class	Number of Tweets
1	3313	1	7385	1	9383
-1	2811	-1	4908	-1	3218
0	796	0	951	0	886

In the datasets from the pandemic period, most of the words in the word cloud were about the COVID-19 pandemic. The most frequently used words are vaccine, viruses, health ministry, and the situation of COVID-19, COVID-19 spread, “Allah”, and a lot of supplications to save them and their families from COVID. Moreover, the distribution of datasets during COVID-19 mentioned in Table 11 demonstrated that the negative sentiments were highest in all cities (Riyadh, Jeddah, and Dammam) compared with other sentiments. It is clear that the COVID-19 epidemic affected a lot of people in Saudi Arabia. As a result, anxiety about COVID-19 events was clearly present in their tweets.

Table 11. Distribution of sentiment classes of (during COVID-19) dataset.

Dammam File (during COVID-19)		Jeddah File (during COVID-19)		Riyadh File (during COVID-19)	
Sentiment Class	Number of Tweets	Sentiment Class	Number of Tweets	Sentiment Class	Number of Tweets
1	2228	1	4111	1	5811
−1	3473	−1	9688	−1	10,418
0	3925	0	7169	0	9713

4.4. Further Discussion

In this section, we compared the effectiveness of the proposed model for Arabic sentiment analysis with some of the previous studies. The previous studies are selected according to multiple criteria: the datasets feature the Arabic language; the classifiers are deep learning algorithms; the source of datasets is Twitter; and the field of datasets is the COVID-19 pandemic.

Considering the results in Table 12, the results demonstrated that both CNN and BiLSTM classifiers outperformed the algorithms in previous studies, except for the LSTM algorithm in [32]. The authors in [32] collected 20,000 tweets while our dataset had 90,187 tweets; thus, there are big differences in the sizes of the datasets. Therefore, the sizes of the datasets and the different classifiers impact the results. In [37], the author achieved good results by using an ensemble deep learning algorithm, but did not use feature extraction techniques that may have increased the accuracy of her model. Furthermore, the authors in [41] study the sentiments of tweets on Saudi trending hashtags, while our work has studied the sentiment of tweets in three specified regions of Saudi Arabia. Further, the authors in [41] used BiLSTM and GRU algorithms, and the result of the BiLSTM algorithm has a 77% F-score. When compared to the result from our work, the BiLSTM in our work outperformed the BiLSTM in [41], with 92% in terms of the F-score.

Table 12. Comparison between Proposed CNN and BiLSTM with Previous Studies in the same Fields (COVID-19 Pandemic).

Ref	Dataset Source and Language	Datasets Size	Feature Used	Algorithm	The Best Result
[32]	Arabic Tweets	30,000 Tweets After preprocessing 20,000 tweets	TFIDF and Word Embedding	LSTM and NB	LSTM: 98.9% accuracy
Proposed work	Arabic Tweets	157,214 Tweets After preprocessing 90,187 Tweets	Word Embedding	BiLSTM, CNN	CNN: 93% accuracy BiLSTM: 92% accuracy
[37]	Arabic Tweets	10 M Tweets After preprocessing 416,292 Tweets	—	SGRUs, SBGRU AraBERT, and the Ensemble deep learning algorithm	Ensemble deep learning algorithm: 90% accuracy
[41]	Arabic Tweets	707,829 Tweets After preprocessing 8000 Tweets	Word Embedding	BiLSTM and GRU	GRU: 81% in terms of F1 score

5. Conclusions

Globally, COVID-19 spreads quickly and affects the health system. It is a worldwide issue and a severe challenge to public health systems and governments globally. During the COVID-19 pandemic, numerous people and government institutions employed social media platforms to exchange information, news, and opinions about this disaster.

This paper studied the Arabic sentiment analysis of Saudi Arabia related to the COVID-19 pandemic. The primary aim of this study is to build an annotated dataset for

analyzing Arabic sentiment for the COVID-19 pandemic and compare the analysis of Arabic sentiments before and after the COVID-19 pandemic. Thus, the datasets have been gathered from Twitter in two different periods. The first period collects the tweets before COVID-19, while the second period collects the tweets during COVID-19. The datasets were preprocessed, filtered, and annotated into three classes: 1 (positive), 0 (neutral), and -1 (negative). After visualizing the datasets, we noticed that the negative sentiments were more common than before the spread of the COVID-19 pandemic.

In this paper, a comparison between many Arabic word embedding techniques was applied. Then, the best word embedding technique was selected for the algorithms as a first layer for deep learning algorithms. The AraVec-SG-300-3 word2vec model has the best result compared to the other word embedding models with different models and different datasets. The deep learning algorithms, CNN and BiLSTM, were used for analyzing Arabic sentiment. Moreover, the outcomes showed that the CNN algorithm is superior to BiLSTM, with 93% accuracy. Hence, the CNN model outperformed the BiLSTM by approximately 1%. So, a comparison has been made between our work and previous works in the same field of Arabic sentiment analysis related to COVID-19. The performance of this research was one of the studies that worked well and demonstrated good performance.

In the future, we intend to compare traditional machine learning and deep learning algorithms on the same dataset. We also recommend applying ensemble deep learning algorithms to enhance the results on the Arabic datasets. In addition, in accordance with the high-quality results [32], we intend to use TFIDF and word embedding as features extraction to improve the results. Other deep learning models should also be investigated for better results, such as [68].

Author Contributions: Conceptualization, methodology, proofreading and supervision, A.R.; implementation, analyses and draft, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sarkodie, S.A.; Owusu, P.A. Investigating the cases of novel coronavirus disease (COVID-19) in China using dynamic statistical techniques. *Heliyon* **2020**, *6*, e03747. [CrossRef] [PubMed]
2. Iacus, S.M.; Natale, F.; Santamaria, C.; Spyrtatos, S.; Vespe, M. Estimating and projecting air passenger traffic during the COVID-19 coronavirus outbreak and its socio-economic impact. *Saf. Sci.* **2020**, *129*, 104791. [CrossRef]
3. Rosenberg, H.; Syed, S.; Rezaie, S. The Twitter pandemic: The critical role of Twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic. *Can. J. Emerg. Med.* **2020**, *22*, 418–421. [CrossRef] [PubMed]
4. Malla, S.J.; Alphonse, P.J.A. COVID-19 outbreak: An ensemble pre-trained deep learning model for detecting informative tweets. *Appl. Soft Comput.* **2021**, *107*, 107495. [CrossRef] [PubMed]
5. Pang, B.; Lee, L. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.* **2008**, *2*, 1–135. [CrossRef]
6. Internet World Users by Language Top 10 Languages. Available online: <https://www.internetworldstats.com/stats7.htm> (accessed on 2 July 2022).
7. Hammad, M.; Al-awadi, M. Sentiment Analysis for Arabic Reviews in Social Networks Using Machine Learning. In *Information Technology: New Generations*; Latifi, S., Ed.; Springer International Publishing: Cham, Switzerland, 2016; pp. 131–139.
8. Abdelminaam, D.S.; Neggaz, N.; Gomaa, I.A.E.; Ismail, F.H.; Elsayy, A.A. Arabicdialects: An efficient framework for Arabic dialects opinion mining on twitter using optimized deep neural networks. *IEEE Access* **2021**, *9*, 97079–97099. [CrossRef]
9. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
10. Alharbi, A.; Kalkatawi, M.; Taileb, M. Arabic Sentiment Analysis Using Deep Learning and Ensemble Methods. *Arab. J. Sci. Eng.* **2021**, *46*, 8913–8923. [CrossRef]

11. Musleh, D.A.; Alkhales, T.A.; Almakki, R.A.; Alnajim, S.E.; Almarshad, S.K.; Alhasaniah, R.S.; Aljameel, S.S.; Almuqhim, A.A. Twitter arabic sentiment analysis to detect depression using machine learning. *Comput. Mater. Contin.* **2022**, *71*, 3463–3477. [[CrossRef](#)]
12. Alharbi, L.M.; Qamar, A.M. Arabic Sentiment Analysis of Eateries' Reviews: Qassim region Case study. In Proceedings of the 2021 IEEE 4th National Computing Colleges Conference, NCCC 2021, Taif, Saudi Arabia, 27–28 March 2021. [[CrossRef](#)]
13. Alsalman, H. An Improved Approach for Sentiment Analysis of Arabic Tweets in Twitter Social Media. In Proceedings of the 2020 3rd International Conference on Computer Applications and Information Security: 0–3, Riyadh, Saudi Arabia, 19–21 March 2020. [[CrossRef](#)]
14. Alyami, S.N.; Olatunji, S.O. Application of Support Vector Machine for Arabic Sentiment Classification Using Twitter-Based Dataset. *J. Inf. Knowl. Manag.* **2020**, *19*, 1–13. [[CrossRef](#)]
15. Almouzini, S.; Khemakhem, M.; Alageel, A. Detecting Arabic Depressed Users from Twitter Data. *Procedia Comput. Sci.* **2019**, *163*, 257–265. [[CrossRef](#)]
16. Elshakankery, K.; Ahmed, M.F. HILATSA: A hybrid Incremental learning approach for Arabic tweets sentiment analysis. *Egypt. Informatics J.* **2019**, *20*, 163–171. [[CrossRef](#)]
17. Al-Smadi, M.; Al-Ayyoub, M.; Jararweh, Y.; Qawasmeh, O. Enhancing Aspect-Based Sentiment Analysis of Arabic Hotels' reviews using morphological, syntactic and semantic features. *Inf. Process. Manag.* **2019**, *56*, 308–319. [[CrossRef](#)]
18. Maghfour, M.; Elouardighi, A. *Standard and Dialectal Arabic Text Classification for Sentiment Analysis*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; Volume 11163, ISBN 9783030008550. [[CrossRef](#)]
19. Oussous, A.; Benjelloun, F.Z.; Lahcen, A.A.; Belfkih, S. ASA: A framework for Arabic sentiment analysis. *J. Inf. Sci.* **2020**, *46*, 544–559. [[CrossRef](#)]
20. Hazım, W.; Gwad, G.; Mahmood, I.; Ismael, I.; Gültepe, Y. Twitter Sentiment Analysis Classification in the Arabic Language using Long Short-Term Memory Neural Networks. *Int. J. Eng. Adv. Technol.* **2020**, *9*, 235–239. [[CrossRef](#)]
21. Ombabi, A.H.; Ouarda, W.; Alimi, A.M. Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks. *Soc. Netw. Anal. Min.* **2020**, *10*, 53. [[CrossRef](#)]
22. Al Omari, M.; Al-Hajj, M.; Sabra, A.; Hammami, N. Hybrid CNNs-LSTM Deep Analyzer for Arabic Opinion Mining. In Proceedings of the 2019 6th International Conference on Social Networks Analysis, Management and Security, Granada, Spain, 22–25 October 2019; pp. 364–368. [[CrossRef](#)]
23. Al-Dabet, S.; Tedmori, S. Sentiment analysis for Arabic language using attention-based simple recurrent unit. In Proceedings of the 2019 2nd International Conference on New Trends in Computing Sciences, Amman, Jordan, 9–11 October 2019. [[CrossRef](#)]
24. Baali, M.; Ghneim, N. Emotion analysis of Arabic tweets using deep learning approach. *J. Big Data* **2019**, *6*, 89. [[CrossRef](#)]
25. Abu Kwaik, K.; Saad, M.; Chatzykiariakidis, S.; Dobnik, S. LSTM-CNN Deep Learning Model for Sentiment Analysis of Dialectal Arabic. In *Arabic Language Processing: From Theory to Practice*; Springer: Cham, Switzerland, 2019; Volume 1108. [[CrossRef](#)]
26. Mohammed, A.; Kora, R. Deep learning approaches for Arabic sentiment analysis. *Soc. Netw. Anal. Min.* **2019**, *9*, 52. [[CrossRef](#)]
27. Omara, E.; Ismail, N.; Network, L. Deep Convolutional Network For Arabic sentiment Analysis. In Proceedings of the 2018 International Japan-Africa Conference on Electronics, Communications and Computations (JAC-ECC), Alexandria, Egypt, 17–19 December 2018; IEEE: New York, NY, USA; pp. 155–159.
28. Abdullah, M.; Hadzikadicy, M.; Shaikhz, S. SEDAT: Sentiment and Emotion Detection in Arabic Text Using CNN-LSTM Deep Learning. In Proceedings of the 17th IEEE International Conference on Machine Learning and Applications, Orlando, FL, USA, 17–20 December 2018; pp. 835–840. [[CrossRef](#)]
29. Heikal, M.; Torki, M.; El-Makky, N. Sentiment Analysis of Arabic Tweets using Deep Learning. *Procedia Comput. Sci.* **2018**, *142*, 114–122. [[CrossRef](#)]
30. Alayba, A.M.; Palade, V.; England, M.; Iqbal, R. A combined CNN and LSTM model for Arabic sentiment analysis. In *Machine Learning and Knowledge Extraction*; Springer: Cham, Switzerland, 2018; Volume 11015. [[CrossRef](#)]
31. Alabdulkreem, E. Prediction of depressed Arab women using their tweets. *J. Decis. Syst.* **2021**, *30*, 102–117. [[CrossRef](#)]
32. Alharbi, N.H.; Alkhateeb, J.H. Sentiment Analysis of Arabic Tweets Related to COVID-19 Using Deep Neural Network. In Proceedings of the 2021 International Congress of Advanced Technology and Engineering, Taiz, Yemen, 4–5 July 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–11. [[CrossRef](#)]
33. Rustam, F.; Khalid, M.; Aslam, W.; Rupapara, V.; Mehmood, A.; Choi, G.S. A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis. *PLoS ONE* **2021**, *16*, e0245909. [[CrossRef](#)]
34. Aljameel, S.S.; Alabbad, D.A.; Alzahrani, N.A.; Alqarni, S.M.; Alamoudi, F.A.; Babili, L.M.; Aljaafary, S.K.; Alshamrani, F.M. A sentiment analysis approach to predict an individual's awareness of the precautionary procedures to prevent COVID-19 outbreaks in Saudi Arabia. *Int. J. Environ. Res. Public Health* **2021**, *18*, 218. [[CrossRef](#)]
35. Jelodar, H.; Wang, Y.; Orji, R.; Huang, S. Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach. *IEEE J. Biomed. Heal. Informatics* **2020**, *24*, 2733–2742. [[CrossRef](#)]
36. Imran, A.S.; Daudpota, S.M.; Kastrati, Z.; Batra, R. Cross-Cultural Polarity and Emotion Detection Using Sentiment Analysis and Deep Learning on COVID-19 Related Tweets. *IEEE Access* **2020**, *8*, 181074–181090. [[CrossRef](#)]
37. Alhumoud, S. Arabic Sentiment Analysis using Deep Learning for COVID-19 Twitter Data. *IJCSNS Int. J. Comput. Sci. Netw. Secur.* **2020**, *20*, 132.

38. Samuel, J.; Ali, G.G.M.N.; Rahman, M.M.; Esawi, E.; Samuel, Y. COVID-19 public sentiment insights and machine learning for tweets classification. *Information* **2020**, *11*, 314. [CrossRef]
39. Xue, J.; Chen, J.; Hu, R.; Chen, C.; Zheng, C.; Su, Y.; Zhu, T. Twitter discussions and emotions about the COVID-19 pandemic: Machine learning approach. *J. Med. Internet Res.* **2020**, *22*, e20550. [CrossRef]
40. Alanezi, M.A.; Hewahi, N.M. Tweets Sentiment Analysis during COVID-19 Pandemic. In Proceedings of the 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy, Sakheer, Bahrain, 26–27 October 2020; IEEE: Piscataway, NJ, USA, 2020. [CrossRef]
41. Alhuri, L.A.; Aljohani, H.R.; Almutairi, R.M.; Haron, F. Sentiment Analysis of COVID-19 on Saudi Trending Hashtags Using Recurrent Neural Network. In Proceedings of the International Conference on Developments in eSystems Engineering, Liverpool, UK, 14–17 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 299–304. [CrossRef]
42. Vyas, P.; Reisslein, M.; Rimal, B.P.; Vyas, G.; Basyal, G.P.; Muzumdar, P. Automated Classification of Societal Sentiments on Twitter With Machine Learning. *IEEE Trans. Technol. Soc.* **2022**, *3*, 100–110. [CrossRef]
43. Rahman, M.M.; Islam, M.N. Exploring the Performance of Ensemble Machine Learning Classifiers for Sentiment Analysis of COVID-19 Tweets. In *Sentimental Analysis and Deep Learning*; Shakya, S., Balas, V.E., Kamolphiwong, S., Du, K.-L., Eds.; Springer: Singapore, 2022; pp. 383–396.
44. Twitter. Twitter Website. Available online: <https://twitter.com/> (accessed on 20 November 2022).
45. OSINT Team. Twint Tool. 2022. Available online: <https://github.com/twintproject/twint> (accessed on 19 February 2022).
46. Wahbeh, A.; Al-Kabi, M.; Al-Radaideh, Q.; Al-Shawakfa, E.; Alsmadi, I. The Effect of Stemming on Arabic Text Classification. *Int. J. Inf. Retr. Res.* **2011**, *1*, 54–70. [CrossRef]
47. Aldayel, H.K.; Azmi, A.M. Arabic tweets sentiment analysis-A hybrid scheme. *J. Inf. Sci.* **2016**, *42*, 782–797. [CrossRef]
48. Altaher, A. Hybrid approach for sentiment analysis of Arabic tweets based on deep learning model and features weighting. *Int. J. Adv. Appl. Sci.* **2017**, *4*, 43–49. [CrossRef]
49. Abuaiadah, D.; Rajendran, D.; Jarrar, M. Clustering Arabic tweets for sentiment analysis. In Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications, Hammamet, Tunisia, 30 October–3 November 2017; pp. 449–456. [CrossRef]
50. Motazsaad. Light Stemmer Algorithm. 2022. Available online: <https://github.com/motazsaad/arabic-light-stemmer> (accessed on 1 March 2022).
51. Yu, T.; Hidey, C.; Rambow, O.; McKeown, K. Leveraging Sparse and Dense Feature Combinations for Sentiment Classification. *arXiv* **2017**, arXiv:1708.03940.
52. Giatsoglou, M.; Vozalis, M.G.; Diamantaras, K.; Vakali, A.; Sarigiannidis, G.; Chatzisavvas, K.C. Sentiment analysis leveraging emotions and word embeddings. *Expert Syst. Appl.* **2017**, *69*, 214–224. [CrossRef]
53. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. In Proceedings of the 1st International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013.
54. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*; Association for Computational Linguistics: Valencia, Spain, 2017; pp. 427–431. [CrossRef]
55. Hosomi, N.; Sakti, S.; Yoshino, K.; Nakamura, S. Deception Detection and Analysis in Spoken Dialogues based on FastText. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Honolulu, HI, USA, 12–15 November 2018; pp. 139–142. [CrossRef]
56. Kim, Y. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Doha, Qatar, 2014; pp. 1746–1751. [CrossRef]
57. Ombabi, A.H.; Lazzez, O.; Ouarda, W.; Alimi, A.M. Deep Learning Framework based on Word2Vec and CNN for Users Interests Classification. In Proceedings of the 2017 Sudan Conference on Computer Science and Information Technology (SCCSIT), Elnihood, Sudan, 17–19 November 2017.
58. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
59. Elzayady, H.; Badran, K.M.; Salama, G.I. Arabic Opinion Mining Using Combined CNN-LSTM Models. *Int. J. Intell. Syst. Appl.* **2020**, *12*, 25–36. [CrossRef]
60. Wang, J. An LSTM Approach to Short Text Sentiment Classification with Word Embeddings. In Proceedings of the 30th Conference on Computational Linguistics and Speech Processing (ROCLING 2018), Hsinchu, Taiwan, 4–5 October 2018; pp. 214–223.
61. Das, S.; Das, D.; Kolya, A.K. Sentiment classification with GST tweet data on LSTM based on polarity-popularity model. *Sadhana Acad. Proc. Eng. Sci.* **2020**, *45*, 140. [CrossRef]
62. Luo, F.-L. *Machine Learning for Future Wireless Communications*; Wiley-IEEE Press: Chichester, UK, 2020; p. 2546, ISBN 9781119562252.
63. Khalil, E.A.H.; El Houbay, E.M.F.; Mohamed, H.K. Deep Learning Approach in Sentiment Analysis: A Review. In Proceedings of the 15th International Conference on Computer Engineering and Systems, Cairo, Egypt, 15–16 December 2020. [CrossRef]
64. Schmidhuber, J. Deep Learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef]
65. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [CrossRef]
66. Soliman, A.B.; Eissa, K.; El-Beltagy, S.R. AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. *Procedia Comput. Sci.* **2017**, *117*, 256–265. [CrossRef]

-
67. Fouad, M.M.; Mahany, A.; Aljohani, N.; Abbasi, R.A.; Hassan, S.U. ArWordVec: Efficient word embedding models for Arabic tweets. *Soft Comput.* **2020**, *24*, 8061–8068. [[CrossRef](#)]
 68. FastText. 2022. Available online: <https://fasttext.cc/docs/en/crawl-vectors.html> (accessed on 13 September 2022).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.