Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.038368
Article

Analyzing Arabic Twitter-Based Patient Experience Sentiments Using Multi-Dialect Arabic Bidirectional Encoder Representations from Transformers

Sarab AlMuhaideb*, Yasmeen AlNegheimish, Taif AlOmar, Reem AlSabti, Maha AlKathery and Ghala AlOlyyan

Department of Computer Science, College of Computer and Information Sciences, King Saud University, P.O. Box 266, Riyadh 11362, Saudi Arabia

*Corresponding Author: Sarab AlMuhaideb. Email: salmuhaideb@ksu.edu.sa

Received: 09 December 2022; Accepted: 10 April 2023; Published: 09 June 2023

Abstract: Healthcare organizations rely on patients' feedback and experiences to evaluate their performance and services, thereby allowing such organizations to improve inadequate services and address any shortcomings. According to the literature, social networks and particularly Twitter are effective platforms for gathering public opinions. Moreover, recent studies have used natural language processing to measure sentiments in text segments collected from Twitter to capture public opinions about various sectors, including healthcare. The present study aimed to analyze Arabic Twitter-based patient experience sentiments and to introduce an Arabic patient experience corpus. The authors collected 12,400 tweets from Arabic patients discussing patient experiences related to healthcare organizations in Saudi Arabia from 1 January 2008 to 29 January 2022. The tweets were labeled according to sentiment (positive or negative) and sector (public or private), and thereby the Hospital Patient Experiences in Saudi Arabia (HoPE-SA) dataset was produced. A simple statistical analysis was conducted to examine differences in patient views of healthcare sectors. The authors trained five models to distinguish sentiments in tweets automatically with the following schemes: a transformer-based model fine-tuned with deep learning architecture and a transformer-based model fine-tuned with simple architecture, using two different transformer-based embeddings based on Bidirectional Encoder Representations from Transformers (BERT), Multi-dialect Arabic BERT (MARBERT), and multilingual BERT (mBERT), as well as a pre-trained word2vec model with a support vector machine classifier. This is the first study to investigate the use of a bidirectional long short-term memory layer followed by a feedforward neural network for the fine-tuning of MARBERT. The deep-learning fine-tuned MARBERT-based model—the authors' best-performing model—achieved accuracy, micro-F1, and macro-F1 scores of 98.71%, 98.73%, and 98.63%, respectively.

Keywords: Sentiment analysis; patient experience; healthcare; Twitter; MARBERT; bidirectional long short-term memory; support vector machine; transformer-based learning; deep learning

1 Introduction

Healthcare organizations are currently shifting to a more patient-centered approach to care, and most modern hospitals rely solely on traditional survey methods—either online or offline—to gather patients' opinions and satisfaction rates regarding the services that the hospitals provide [1]. However, although traditional surveys are easy to develop, they cannot capture patients' true feelings and observations. Patients do not usually feel motivated to provide honest and accurate answers, thereby lowering the response rate and validity of surveys [2]. Therefore, there is a need for a better means of patient engagement and empowerment.

Several studies have examined the impact and potential of social media in the healthcare sector [3–5], and Twitter has emerged as a major channel for expressing feelings, thoughts, and experiences with a community that shares similar interests and values [6]. Twitter provides a huge amount of raw and unsolicited opinions from patients about the services provided by healthcare organizations, which can be used as an alternative to traditional patient experience surveys. In Saudi Arabia, Twitter is the third most visited site after Google and YouTube as of 2021, making it the second most visited social media site [7]. However, while there are several available corpora representing tweets in the Arabic language related to general topics (e.g., the Saudi Dialects Corpus from Twitter (SDCT) in the Saudi dialect [8], the Arabic Sentiment Tweets Dataset (ASTD) in the Egyptian dialect [9], and several in multiple dialects, such as the Arabic Speech Act and Sentiment corpus of tweets (ArSAS) [10] and the Arabic Sarcasm detection dataset (ArSarcasm-v2) [11]), there are none exclusively in the healthcare domain. Several studies have analyzed sentiments about diverse topics such as finance [12], government and politics [13,14], economics [15], and entertainment [16,17], but few have explored the possibility of using sentiment analysis (SA) in health-related applications. Moreover, studies that cover the use of SA to measure patient experience and satisfaction rates—in either English or Arabic—are even fewer. This is because of the lack of Arabic corpora for patient experience and the lack of programming tools that support the complex structure of the Arabic language, especially dialectic Arabic.

Although Twitter is a rich source of sentiments and opinions, analyzing these sentiments manually is both time-consuming and expensive. Moreover, the process of classifying dialectic Arabic text—the most generic form of communication on Twitter—could represent a bottleneck due to several factors, including the extensive use of slang, the widespread use of abbreviations, and the neglect of spelling and grammar rules [6]. A more accurate, efficient, and effective approach is needed to replace the existing traditional methods and automate the process of SA to measure patients' satisfaction and analyze their feedback.

At the broadest level, there are two types of SA methods [18]: (i) supervised learning and (ii) lexicon-based methods. Supervised models can also be divided into those based on deep learning (DL) and those based on feature engineering. Feature engineering-based techniques predict sentiment by learning from various features chosen to capture various facets of the text. Meanwhile, DL is regarded as the state of the art in machine learning (ML) and has succeeded greatly in many areas, particularly in computer vision and natural language processing (NLP) [19]. Different neural network (NN) designs are typically trained using embedded representations of text units (characters or words) as input characteristics.

Transformers [20] are a type of NN for learning sequential inputs such as images and natural languages, and they use not recurrent connections but rather an architecture that is close to a classical, fully connected one. A transformer's ability for contextual modeling surpasses that of preceding DL approaches such as long short-term memory (LSTM) [20–22]. The main feature that distinguishes transformers from other models is the use of self-attention layers [20,23], which provide any processed input with its relevant context, and the processing can be performed in any order. Thus, transformers are massively parallelizable, making them faster than other recurrent NN (RNN) [24] models where the processing must be performed in a single direction [20]. Another advantage of transformers is that once they are pre-trained in an unsupervised manner on a large amount of textual data, the model can be fine-tuned to a specific downstream task with relatively few labels by simply adding another layer after the last transformer layer and training the entire network for only a few epochs. Because the general linguistic patterns have already been learned during pre-training, the fine-tuning process is efficient and effective [22,23]. Transformer-based language models (LMs) based on Bidirectional Encoder Representations from Transformers (BERT)—such as multilingual BERT (mBERT), the BERT transformer model for Arabic language (AraBERT), and Multi-dialect Arabic BERT (MARBERT) [22,25]—have recently advanced the state of the art for Arabic SA and NLP [26].

In the study reported herein, the authors used the recent advances in transformer-based ML to train an ML classification model using a dataset of Arabic patients' experiences collected from Twitter that were labeled manually as positive or negative. The contributions of this work are as follows. First, this study was aimed at bridging the gap in medical-related applications of SA and introducing an Arabic patient experience corpus: Hospital Patient Experiences in Saudi Arabia (HoPE-SA). HoPE-SA is the first dataset of its kind that explores carefully curated data on patient experiences in Arabic from the Twitter platform. The newly constructed open-source HoPE-SA dataset is aimed at bridging the gap in Arabic NLP resources and corpora, especially in the health sector. It could also motivate and encourage researchers and developers to conduct further studies that might significantly improve Arabic NLP and the health sector. Moreover, healthcare organizations can benefit from this study using trained models to measure the level of patient satisfaction and accordingly improve their provided services to better adhere to patient needs and achieve a more patient-oriented care system. Second, the authors applied simple statistical analysis to the collected dataset to examine the patients' sentiments about the healthcare services provided in Saudi Arabia, as well as to investigate whether there is a tangible difference in the patient sentiments about services provided by the private and public (governmental) healthcare sectors. Lastly, the authors trained five models to distinguish sentiments in tweets automatically with the following schemes: a transformer-based model fine-tuned with a DL architecture, a transformer-based model fine-tuned with a simple architecture, and a pre-trained word2vec model with a support vector machine (SVM) classifier, using two different transformer-based embeddings, MARBERT [25] and mBERT [22].

MARBERT is intended to condition jointly on both the left and right context in all layers to pre-train deep bidirectional representations from the unlabeled text. As a result, the pre-trained MARBERT model may be fine-tuned with an additional output layer to produce models for various tasks, including SA, emotion and sarcasm detection, topic classification, dialect identification, named entity recognition, and question answering [22,25]. Notably, this is the first study to investigate the fine-tuning of MARBERT using a bidirectional LSTM (BiLSTM) layer. The closest related work is that of Nguyen et al. [27] for a Vietnamese dataset, where mBERT was fine-tuned separately by several methods, and that of Souza et al. [28], who reported on Portuguese BERT models within different architectures, aiming to classify named entities in the Portuguese language, as detailed in Section 2.4. The present work drew inspiration from those two previous studies but is novel.

The authors aimed to answer the following research questions. (1) What is the attitude of patients in Saudi Arabia toward the provided healthcare services? (2) Is there a tangible difference in the quality of services provided by private and public healthcare sectors from the perspective of their patients? (3) Does fine-tuning MARBERT by adding a BiLSTM layer provide better performance? Does fine-tuning it with a simple feedforward NN (FNN) layer provide better performance? In other words, does extracting more features using a BiLSTM layer improve the performance?

The rest of this paper is organized as follows. Section 2 surveys the literature for the state of the art in Arabic SA, including SA for patient experience. Section 3 provides a detailed description of the process of collecting, annotating, preprocessing, analyzing, modeling, and classifying Twitter-based Arabic sentiments about patients' experiences. The results obtained are presented and evaluated in Section 4. The findings are discussed in Section 5, and the paper concludes in Section 6.

2 Related Work

Although SA has several applications in different domains, its role in improving the healthcare sector remains understudied. Moreover, few studies have applied SA to patient experiences, especially in the Arabic domain (Section 2.1). For the word embeddings or LM, three categories can be identified. The first includes frequency-based LMs such as a bag of words (BoW), term frequency–inverse document frequency (TF-IDF), and *n*-gram models [29,30]. The second includes prediction-based LMs such as word2vec, which can be implemented using two approaches: continuous BoW (CBoW) and skip-grams [31–33]. The third includes transformer-based LMs [34]. A combination of three different LMs has also been used [35]. Several studies have proposed different approaches for analyzing Arabic sentiment from social content, including modern standard Arabic (MSA) and dialectic Arabic (DA). In addition, because sarcasm, emotion detection, and named entity recognition are specific cases of SA [36], their related studies are also considered. Thus, the related studies are classified according to whether the proposed approach was based on a classical ML method (Section 2.2), a DL method (Section 2.3), or a transformer-based learning model (Section 2.4). Also emphasized are studies conducted using BERT for feature representation and transfer learning. Some of the reviewed work involved lexicon-based classification [37]. Note that all the reviewed studies targeted Arabic SA except those by Liu et al. [38] (English), Souza et al. [28] (Portuguese), and Nguyen et al. [27] (Vietnamese). Finally, the findings are discussed in Section 2.5.

2.1 Sentiment Analysis for Patient Experience

For classifiers, Alayba et al. [39] used three classical approaches (SVM, logistic regression, and naïve Bayes) and two DL approaches [convolutional NNs (CNNs) and a deep NN]. They introduced their dataset of Arabic tweets expressing opinions about health services, comprising 2026 tweets; the dataset was annotated manually. For feature representation, a combination of unigram, bigram, and TF-IDF methods was used, and the highest accuracy was achieved using the SVM classifier. Alayba et al. focused on providing the dataset rather than analyzing the results. Liu et al. [38] harnessed ML and transfer learning to study the tendencies of public opinions, attitudes, and behaviors regarding COVID-19 vaccines. For the ML models, SVM, random forest, and logistic regression were used with TF-IDF encoding. As for transfer learning, they used BERT [22] as a classifier and LM, and the transfer learning model outperformed the other ML models.

2.2 Classical Machine Learning Approaches for Sentiment Analysis

Bayazed et al. [8] introduced the open-source dataset SDCT. The features were extracted using the TF-IDF weighting scheme and an n-gram model, then 11 ML classifiers were trained using the extracted features. The results indicated that there was no significant difference among unigrams, bigrams, and trigrams, especially for short texts such as tweets. Aldayel et al. [6] presented a hybrid lexicon–SVM sentiment classifier, used along with three n-gram models and TF-IDF as feature vectors. Aljameel et al. [40] analyzed the sentiments of tweets about several COVID-19 preventive measures conducted in Saudi Arabia in an attempt to understand the awareness of citizens about these measures. They used an n-gram feature extraction along with several classical ML methods, such as SVM, k-nearest neighbors, and naïve Bayes classifiers. The highest accuracy was achieved by the SVM classifier along with the bigram model.

2.3 Deep Learning Approaches for Sentiment Analysis

Alahmary et al. [41] performed SA on Saudi dialect tweets using LSTM and its variant BiLSTM; they used the SDCT as the dataset, and the word embeddings were generated using the CBoW word2vec model. In another study, Dahou et al. [42] trained a word embedding algorithm on their collected corpus and a CNN for classification. CBoW and skip-gram were used for the word embeddings. They compared the accuracy of the equally balanced and unbalanced CNN model with the results from three previous studies and applied it to nine different public datasets. They found that training with an unbalanced dataset resulted in higher accuracy. A CNN and BiLSTM were also used by Heikal et al. [43] for SA on Arabic tweets, in addition to a third hybrid model that combined the two deep learners. The tweets' word embeddings were generated by the Arabic word2vec skip-gram model (AraVec) [44], which was trained using Arabic tweets. The ensemble model determined the class using soft voting and had the best performance in terms of accuracy and F1 score.

2.4 Transformer-Based Approaches for Sentiment Analysis

Abdelali et al. [45], the authors behind Qatar Computing Research Institute Arabic and Dialectal BERT (QARiB), pre-trained five different BERT models on Arabic tweets and newspapers. They noted that an increase in the number of training steps or the use of more datasets did not necessarily result in a better model. They also concluded that using MSA and informal datasets to train the model helped, even if the model was only meant to be used on informal data. Bashmal and AlZeer examined an ensemble BERT model to detect sarcasm in Arabic tweets [46] using the Arabic sarcasm detection (ArSarcasm-v2) dataset [11], and the full model had a better F1 score in comparison to the constituent models alone. Chouikhi et al. [47] showed an interesting approach to using a BERT model for Arabic by tackling the tokenization problem instead of merely the training and fine-tuning phases.

Abdul-Mageed et al. [25] argued that multilingual LMs such as the cross-lingual LM robustly optimized BERT pre-training approach (XLM-RoBERTa) [48] and mBERT [22] can be easily outperformed by monolingual models that are pre-trained with larger and more language-specific datasets. They also argued that some existing LMs do not perform well in real-world settings, such as social media, because they were trained on datasets that do not capture the informality and diversity of the language in social media. Abdul-Mageed et al. presented two novel Arabic transformer-based LMs, Arabic BERT (ARBERT) and MARBERT, pre-trained on a large, diverse dataset of DA and MSA. In another study, Abu Farha et al. [49] used the ArSarcasm-v2 dataset [11] to train different transformer-based models, and a BiLSTM model was used as a baseline. It was found that AraBERT large [50] and MARBERT [25] had the highest F1 scores for the positive and negative classes. Abu Farha et al.

concluded that models with more parameters could potentially obtain more representational power, thus performing better than the smaller variants. The competitiveness of AraBERT and MARBERT was also shown by Naski et al. [51].

The effects of different fine-tuning approaches on a pre-trained BERT model were compared by Nguyen et al. [27], who performed SA on two Vietnamese datasets. The fine-tuning methods included standard fine-tuning, where an additional layer was appended, and a more complex approach, where the whole output sequence was fed to another classification model. The classification models were LSTM, text-based convolutional NN (TextCNN) [52,53], which is a widely used CNN for language-related tasks, and a recurrent convolutional NN (RCNN) [54]. All the fine-tuning was performed on a multilingual BERT model, and upon evaluation, the best-performing models were the fine-tuned RCNN followed by the fine-tuned TextCNN. Similarly, Souza et al. [28] reported BERT models in different architectures, aiming to classify named entities in Portuguese. Four BERT variants were studied, i.e., (i) BERT with a last-layer BiLSTM classifier, (ii) BERT with a last-layer BiLSTM classifier and a conditional random field (CRF) layer [55], (iii) BERT whose layer weights were all adjusted and with an added simple linear classifier layer, and (iv) the same scheme but with an extra CRF layer. The Portuguese BERT large with the simple linear classifier resulted in the highest F1 score, and the CRF layer did not have a significant impact on the later model.

2.5 Discussion of Previous Approaches

Except for QARiB [45], only minimal preprocessing is required when using transformer-based classifiers [49,51,55], while data cleaning and normalization have been used frequently in other studies that applied classical and DL approaches and embeddings. Furthermore, Al-Twairesh [56] compared different AraVec [44] models, such as CBoW (100, 300) and skip-gram (100, 300), and it was found that the skip-gram model with 300 embeddings outperformed the other models. As suggested by Liu et al. [38], Bashmal et al. [46], and Al-Twairesh [56], the transformer-based LM BERT is superior to frequency-based and prediction-based embeddings, which is because of several factors. The first is the transformers' ability to learn better contextual embeddings. The second is that once pre-trained, word2vec only generates a single static embedding for a given word, while in BERT the process of word embedding generation is dynamic and depends on the given context.

Several studies applied different sentiment classification approaches. Based on the studies by Liu et al. [38], Alahmary et al. [41], and Al-Twairesh [56], the present authors conclude that classical ML models with frequency-based or prediction-based feature representations result in lower performance measures when compared to DL and transformer-based approaches. Additionally, transformer-based models have outperformed classical and DL models when applied with the same settings. The multi-dialect BERT reported by Al-Twairesh [56] performed worse than other models within the same study. However, MARBERT, which was also pre-trained on multi-dialect Arabic tweets, attained significantly higher performances in the work by Abdul-Mageed et al. [25], Abu Farha et al. [49], and Naski et al. [51]. This difference in performance was a result of the size of the dataset. Al-Twairesh [56] used a dataset that was relatively small compared to the dataset used by Abdul-Mageed et al. [25], with ten million tweets and one billion tweets, respectively. MARBERT has outperformed AraBERT and ARBERT in SA [25]. This is because, unlike AraBERT and ARBERT, MARBERT was pre-trained on DA instead of MSA. Moreover, fine-tuning pre-trained models consumes fewer resources compared to other ML approaches.

3 Materials and Methods

This section illustrates the process of analyzing Twitter-based Arabic sentiments regarding patients' experiences by specifying the techniques used in the dataset collection (Section 3.1), dataset annotation (Section 3.2), statistical analysis (Section 3.3), and dataset preprocessing (Section 3.4), as well as specifying the LMs, classification models, and model evaluation procedures (Section 3.5). Finally, the SVM and BERT-based baseline models are described (Section 3.6). The workflow is described briefly as follows. First, the dataset was collected from Twitter and was annotated manually by sentiment and the healthcare sector. The authors then performed statistical analysis on the collected dataset and preprocessed it. For the statistical analysis, the authors compared patients' levels of satisfaction with public and private healthcare organizations. Meanwhile, for preprocessing, the dataset was cleaned and normalized, and then each tweet was tokenized before being fed into the LM, which produced the word embeddings. The classification model was subsequently trained from the annotated dataset and was able to sentimentally classify any fed text into positive or negative. To evaluate the model, 10-fold cross-validation was used. The following subsections describe the workflow in greater detail, and Fig. 1 summarizes it.

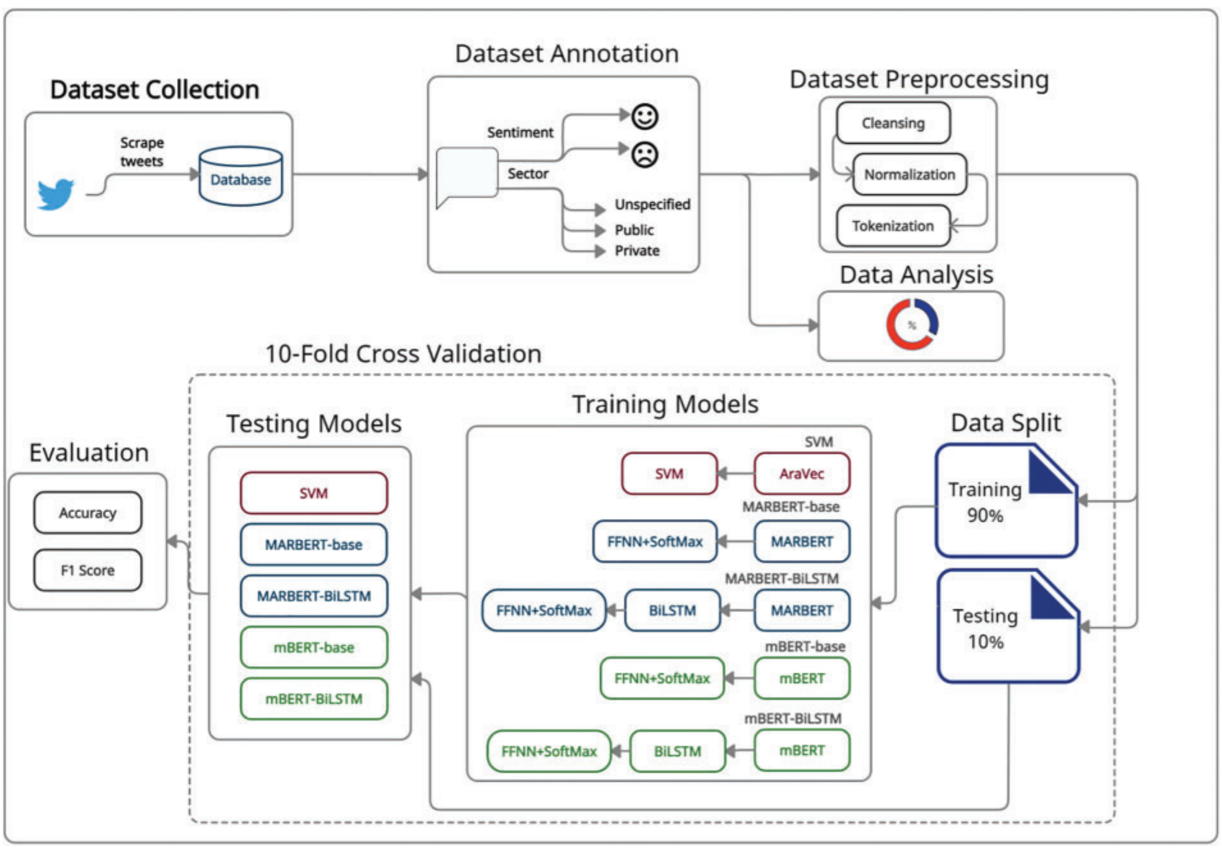

Figure 1: Pipeline of the proposed approach for dataset construction and sentiment analysis of tweets to measure the attitudes of patients toward the services provided by healthcare organizations in Saudi Arabia

3.1 Dataset Collection

The construction of an Arabic patient sentiment dataset was one of the main deliverables of this study. The authors' dataset consists of patient-related Arabic tweets that reflect sentiments toward healthcare organizations in Saudi Arabia. The Twitter intelligence tool (TWINT) [57] library was used to retrieve the raw tweets from Twitter. The authors extracted tweets by specifying the Twitter identifiers for healthcare organizations in Saudi Arabia. Identifiers (mentions) were chosen as the search queries because they—from the authors' observation—tended to return tweets that contained more sentiments and less unwanted noisy data, such as ads, compared to retrieving tweets by using keywords or hashtags. Fourteen healthcare organizations (see Suppl. Table 1 in the Supplementary Material) were selected according to the 35 highest-ranked hospitals in Saudi Arabia [58] after the authors excluded hospitals with inactive Twitter accounts and/or low activity on Twitter and replaced them with hospitals that had a Twitter presence. The authors collected 330,963 raw tweets that were tweeted between 1 January 2008 and 29 January 2022 and contained at least one word. The entire process from data collection to data filtering and annotation took three months, from 10 January 2022 to 6 April 2022. The authors named the dataset HoPE-SA, which stands for Hospital Patient Experiences in Saudi Arabia.

3.2 Dataset Annotation and Filtering

Each tweet was labeled according to its sentiment as positive (+1), negative (−1), or neutral (0), the latter to be excluded. Moreover, each tweet was also labeled as public, private, or unspecified according to the sector of the healthcare organization alluded to in the tweet. It is important to note that the annotation according to sectors was for statistical analysis purposes and was not used in training the classifiers. The annotation process was as follows: each instance of the dataset was annotated by an odd number of annotators (three), and the final label of each instance was determined by the majority vote. The annotators followed the following guidelines inspired by Bayazed et al. [8] and Alahmary et al. [41].

1. If the tweet's content was not related to patient experience (which can be defined as any interaction of patients with healthcare-providing organizations and can be reported by the patients themselves or their relatives), it was marked as irrelevant.
2. If the tweet did not contain sentiments (such as declarations, public announcements, queries, and ads), it was marked as neutral.
3. If the tweet displayed positive sentiments (joy, happiness, satisfaction, gratitude, and other favorable emotions), it was marked as positive. Otherwise, if the displayed sentiments were negative (anger, sadness, disappointment, and other unsatisfactory emotions), it was marked as negative.
4. If the healthcare organization name was mentioned, it was marked as public or private according to the Saudi Central Board for Accreditation of Healthcare Institutions (CBAHI) list [59]; otherwise, it was marked as unspecified.
5. Tweets that were classified as neutral or irrelevant were removed from the dataset, while tweets with unspecified sectors were omitted from the statistical analysis.

Finally, to evaluate the quality of the annotations, the inter-rater agreement of the annotators over the dataset was measured with the Fleiss kappa metric [60], which measures the agreeability of any number of annotators n and classes k with N examples (tweets) while eliminating coincidental agreement:

$$\kappa = \frac{\overline{P} - \overline{P}_e}{1 - \overline{P}_e}, \tag{1}$$

where n_{ij} is the number of annotators who assigned example i the class j, and $\overline{P}$ and $\overline{P}_e$ are calculated as

$$\overline{P} = \frac{1}{Nn\,(n-1)}\left(\sum_{i=1}^{N}\sum_{j=1}^{k} n_{ij}^2 - Nn\right), \tag{2}$$

$$\overline{P}_e = \sum_{j=1}^{k}\left(\frac{1}{Nn}\sum_{i=1}^{N} n_{ij}\right)^2. \tag{3}$$

The calculated value of k was 0.96, which indicates almost perfect agreement among the annotators [61].

3.3 Statistical Analysis

In this study, the authors aimed to derive statistical inferences from the collected dataset about the differences in patients' satisfaction levels between the public and private healthcare sectors. The analysis was based on the sentiments and sectors that were alluded to in the collected tweets. The authors also collected general statistical information about the dataset, such as counts and rates of positive and negative tweets (in terms of sentiments) and the sectors (public and private).

3.4 Dataset Preprocessing

In this study, AraVec and BERT-based architectures were used as feature extraction models. The output of AraVec served as feature input vectors for the SVM classifier, and the output of MARBERT and mBERT served as feature input vectors for the subsequent classification layers that were used to address the sentiment classification. To eliminate noise, unify the tweets, and improve the learning process, several preprocessing techniques were implemented, which can be summarized in two steps: cleaning and normalization. Regular expressions were used to perform the two steps. As for the AraVec word embedding model, note that the authors followed almost the same preprocessing steps as those mentioned by Soliman et al. [44] (the developers of AraVec) to utilize the full potential of the pre-trained AraVec model. The cleaning process varied with the approach (see Table 1 for the differences), but normalization was unified across the different approaches [44]. After cleaning and filtering, the large number of tweets was reduced to 12,400, specifically 7849 negative ones and 4551 positive ones.

Emojis were kept unchanged in MARBERT and mBERT because they had vector representations. However, AraVec was not pre-trained on texts that contained emojis, so it had no vector representation for them. Rather, Soliman et al. [44] (the authors of AraVec) converted each emoji into its overall sentiment, i.e., positive or negative emotion. The present authors used the same technique for emojis in the tweets when training the SVM model; to do so, they used the emoji sentiment mapping described by Hakami et al. [62] to create a Python dictionary that performed this mapping.

Table 1: Cleaning steps used for support vector machine (SVM) and Bidirectional Encoder Representations from Transformers (BERT)-based approaches

Cleaning step	SVM	BERT-based
Remove links and mention tags	✓	✓
Remove punctuation	✓	✓
Remove underscores and hash symbols from hashtags	✓	✓
Remove digits	✓	✓
Remove English words	✓	✓
Remove diacritics	✓	
Remove elongation	✓	
Remove stop words	✓	
Convert emojis to text	✓	

Finally, each tweet was tokenized into individual words. Cleaning and normalization were performed using the re (regular expression) Python library, while tokenization was performed using the Natural Language Tool Kit (NLTK) for the SVM model and BERT's WordPiece tokenizer [26] for MARBERT and mBERT.

3.5 Proposed Learning Approach

After preprocessing, the authors tokenized the tweets and unified each tweet's token vector to a length of 250 tokens, where shorter tweets were padded with a special token [PAD], and the authors appended a predefined class token at the beginning [CLS] and a predefined separator [SEP] at the end of each tweet. The next step was to map each tweet's token vector to its corresponding ID and attention mask. Finally, the authors converted all tokens and attention mask vectors into Torch tensors to make them compatible with the PyTorch framework. After collecting and preprocessing the data, the authors fed them as inputs into a sentiment classification framework to determine their polarity. As for word embeddings, the authors used the Arabic BERT variant MARBERT. Meanwhile, for classification, the authors considered fine-tuning MARBERT. Google Colaboratory (Colab) [63] was used as an environment for the BERT model implementation. The authors' implementations were carried out using the open-source ML framework based on the Python programming language and the Torch library (PyTorch) because of its strong and robust support for graphical processing units (GPUs). Because of the dynamic nature of Colab's resource allocation, the GPUs were not the same in all experiments, but the majority were performed on a single NVIDIA T4 GPU or a single NVIDIA Tesla P100 GPU. The following subsections illustrate the approach in detail.

The authors selected MARBERT [25] because (i) it was trained on different Arabic dialects (not only MSA) with 100,000 Arabic subwords and (ii) its dataset comprises many tweets relevant to the authors' domain of interest. BERT uses WordPiece tokenization [26], which is an effective tokenization method for reducing vocabulary size; instead of treating the different forms of the same word as different tokens, (e.g., walking, walk, and walked), it maps them into one word-piece token (e.g., walk) and learns based on this. This method produces tokens separated by "##" and covers many vocabularies in different languages, including Arabic.

Influenced by the studies of Nguyen et al. [27] and Souza et al. [28], the present authors implemented two main fine-tuning techniques with MARBERT at the core. The first technique used a simple FNN to perform the classification task, while the second technique used a complex NN architecture with layers that performed further feature extractions, followed by an FNN for the classification output. For the complex layer, the authors chose BiLSTM because of its good performance in SA according to Abu Farha et al. [26]. In Nguyen et al. [27], the complex architectures outperformed the FNN architecture by a small margin, whereas in Souza et al. [28] the simpler architecture had better results. Because these two previous studies were focused on Vietnamese and Portuguese, respectively, the present authors investigated whether using Arabic would produce similar outcomes with the same framework. When fine-tuning MARBERT with an FNN layer, the input to the FNN layer was the vector-only [CLS]; the [CLS] token was a special token added at the beginning of sentences, and its output vector was used for classification tasks. Fig. 2 shows the architecture when MARBERT was fine-tuned by adding a complex classification layer—BiLSTM—that performed further feature extraction. In this technique, all the $n + 1$ output vectors were used as inputs to the BiLSTM layer, where n is the length of the sentence plus one for the [CLS] token added at the beginning.

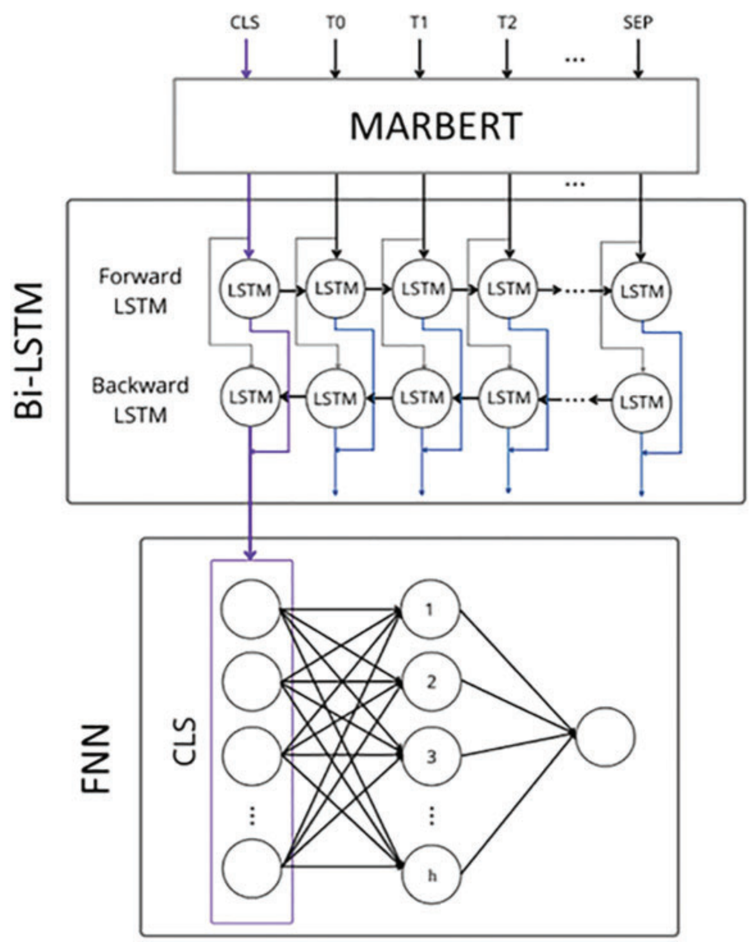

Figure 2: Architecture of MARBERT fine-tuned with bidirectional long-short-term memory–feedforward neural network (BiLSTM-FNN) classification layers

The authors performed a series of experiments to tune MARBERT's hyperparameters, using five-fold instead of 10-fold cross-validation because of the large amount of time that the latter would have required and the low variation in performance between the models. The initial hyperparameters used were a combination of default and recommended values in Devlin et al. [22], described as follows. The authors used an FNN hidden layer size of 50 with a learning rate of 5×10^{-5}, an adaptive moment estimation with decoupled weight decay (AdamW) optimizer, and a cross-entropy loss function. The batch size was set to 32 with two epochs. The hidden layer size for BiLSTM was also initialized to 50. The authors used batched training with gradient norm clipping. Finally, to evaluate the resulting models, the authors used 10-fold cross-validation and computed the average folds' accuracy and F1 score (positive, negative, micro, and macro) measures for performance evaluation.

Let TP and TN denote the numbers of true positives and true negatives, respectively, and FP and FN denote the numbers of false positives and false negatives, respectively. Accuracy is defined as the proportion of correctly predicted examples:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{4}$$

Precision is the fraction of correctly classified positive examples among all positively classified examples:

$$Precision = \frac{TP}{TP + FP}. \tag{5}$$

Meanwhile, recall or sensitivity measures the ratio of correctly classified positive examples to the true positive examples:

$$Recall = \frac{TP}{TP + FN}. \tag{6}$$

The F1 score is calculated as the harmonic mean of the precision and recall; thus, it combines both precision and recall in a single value:

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{7}$$

When considering the F1 scores of a model with respect to positive and negative classes, i.e., $F1^{+}$and $F1^{-}$, respectively, the average performance is used as an indicator of the overall performance. To calculate the macro-F1 score, all values of $F1$ should be added and then averaged:

$$Macro - F1 = \frac{F1^{+} + F1^{-}}{2}. \tag{8}$$

For the micro-F1 score, for each class, the value of $F1$ is weighted with a weight w (w^{+} and w^{-} for positive and negative examples, respectively), which is the ratio of the examples that are represented by this class to all the examples. Then, all values of $F1$ are summed to produce the micro-F1 score:

$$Micro - F1 = w^{+} \times F1^{+} + w^{-} \times F1^{-}. \tag{9}$$

3.6 Baseline Models

According to Nguyen et al. [27], Alayba et al. [39], Alahmary et al. [41], and Al-Twairesh [56], SVM is a simple and powerful approach for SA. The results presented by Nguyen et al. [27] indicated that SVM performs well even compared with DL models or BERT variant models. Accordingly, the present authors selected SVM from all classical ML algorithms because it is a simple and widely used algorithm that has proven its competitiveness in previous work. Other classical ML methods were not considered in this work because they have already been demonstrated in the literature to be incompetent in comparison with the state-of-the-art transformer-based methods [38]. For structural and environmental compatibility and to measure the impact of the MARBERT pre-training dataset on the present fine-tuning, the authors also chose mBERT—the multilingual version of BERT—as another baseline model. The mBERT models had the same structure as the MARBERT models.

3.6.1 Support Vector Machine (SVM) Model

An essential part of SVM is the choice of kernel functions; the linear kernel is a good choice for complicated data [64], but the authors also considered the Gaussian or radial basis function (RBF) kernel. For the word embeddings model, the authors considered the prediction-based embedding AraVec [44]. AraVec is a pre-trained NN representation model that uses Arabic data from Twitter, the World Wide Web, and Wikipedia. Moreover, AraVec was trained via two techniques: CBoW and skip-gram. AraVecSG300, which stands for AraVec skip-gram with 300 features, showed better performance by a margin of 10% compared to the BERT variants, as presented in Al-Twairesh [56].

To train the SVM model, the authors first converted the preprocessed tweets to numerical vectors (word embeddings) using AraVec, a pre-trained version of word2vec with a vocabulary size of 1,476,715 Arabic words. The authors used the Gensim Python library to access the pre-trained model and set the number of features to 300, as recommended by Soliman et al. [44] (the authors of AraVec). AraVec returned a one-dimensional vector of 300 numbers (features) for each word in the tweet. To find a vector that represented the entire tweet, the authors averaged the individual word vectors, as suggested by Khalil et al. [65]. Then, the vector of each tweet was used to train an SVM model using the sci-kit-learn Python library with a regularization parameter of $C = 1.0$ and the squared L2 penalty as the loss function by default. The authors experimented with two kernel functions, i.e., linear and Gaussian RBF functions, as well as two architectures for the AraVec model, i.e., CBoW and skip-gram, resulting in a total of four models as given in Table 2. Finally, 10-fold cross-validation was implemented to evaluate the accuracy and other performance measures. For SVM implementation, the authors used Jupyter Notebook locally on a Windows system running on a computer with an i7 Core CPU and 8 GB of RAM.

Table 2: Summary of parameters used in SVM models

SVM model	Kernel	Word embeddings
Model 1	Linear	AraVecCBoW300∗
Model 2	RBF	AraVecCBoW300
Model 3	Linear	AraVecSG300∗
Model 4	RBF	AraVecSG300

Note: ∗AraVecCBoW300 refers to the pre-trained AraVec model with a continuous bag of words (CBoW) architecture and 300 features per vector, and AraVecSG300 is based on the skip-gram architecture.

3.6.2 Bidirectional Encoder Representations from Transformers (BERT)-Based Baseline Models

For further investigation, the authors implemented another transformer model—multilingual BERT [22], particularly mBERT—with the same fine-tuning procedure and configurations as those used with the MARBERT models. The mBERT model was used to examine MARBERT's performance compared to another model from the same family and to be able to compare the authors' models to other models from various languages. Moreover, because mBERT was pre-trained on Wikipedia articles written mainly in MSA, it could be used to investigate the effect of the pre-training of the dataset on the performance of fine-tuned models in downstream tasks. Additionally, mBERT could also be used to assess the gain from adding the BiLSTM layer by comparing two transformer-based models fine-tuned with and without the BiLSTM layer. mBERT was implemented using Google Colab to benefit from the freely available GPUs.

4 Results

This section is divided into four subsections, each reporting the results obtained with an adopted approach. Section 4.1 reports the findings from the statistical analysis. Section 4.2 reports the results obtained after training the SVM model using AraVec for word embedding extraction. Section 4.3 showcases the hyperparameter tuning phase for the MARBERT models and reports the final results obtained. Finally, Section 4.4 reports the results of the fine-tuned BERT-based models (MARBERT and mBERT); it highlights the differences between the results of each BERT-based model and compares the two different fine-tuning techniques.

4.1 Statistical Analysis

The dataset's tweets contained 150 characters on average, with a minimum of 16 and a maximum of 436, including mentions and links. After removing stop words in the dataset, the most frequent words were مستشفى (hospital) or المستشفى (the hospital), التخصصي (specialist), المرضى (the patients) or المريض (the patient), الدكتور (the doctor), and موعد (appointment). This frequency might have been affected by the keywords that were used in retrieving the tweets. Fig. 3a shows the word cloud of the dataset, and Fig. 3b shows the corresponding word cloud for the tweets translated into English using the deep_translator Python tool.

(a) (b)

Figure 3: Word clouds of HoPE-SA (Hospital Patient Experiences in Saudi Arabia) dataset after removing stop words: (a) original word cloud; (b) corresponding word cloud for English-translated tweets

Moreover, 90% of the tweets were grouped based on the year of tweeting, there being a problem with the date format for the remaining 10%. Fig. 4 shows the results of the statistical analysis for tweets collected in HoPE-SA for the years 2012–2022.

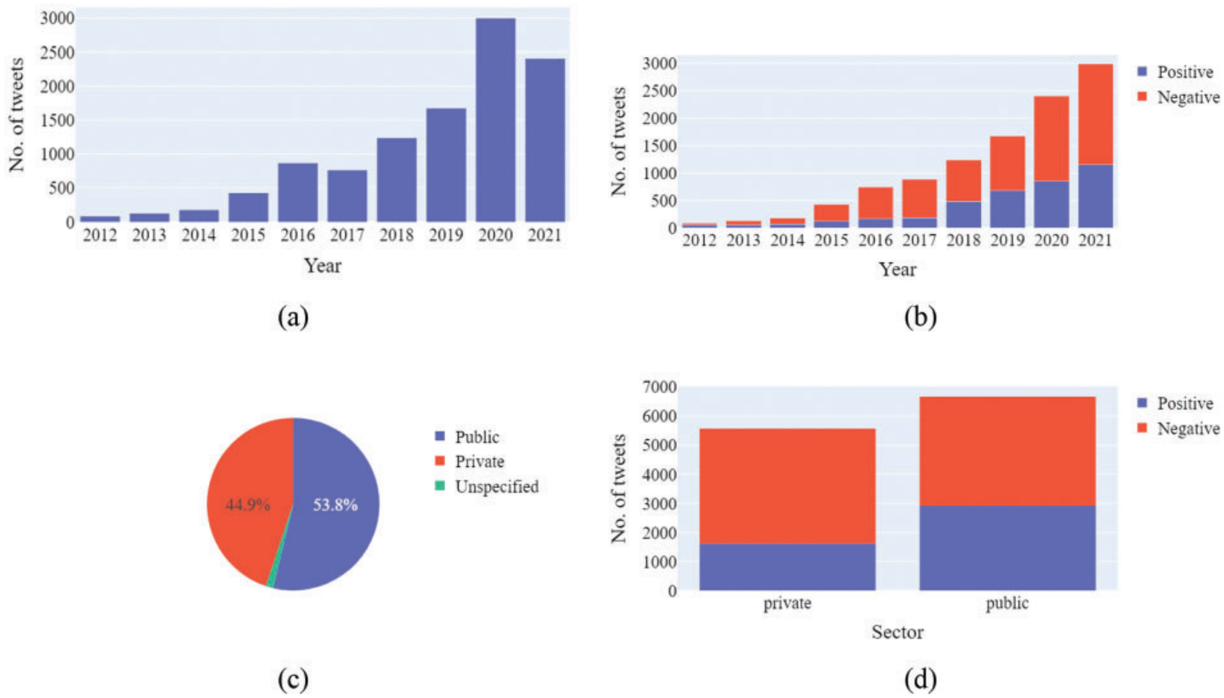

Figure 4: Results of a simple statistical analysis for HoPE-SA: (a) number of tweets per year; (b) number and proportion of positive and negative tweets per year; (c) proportion of tweets for each sector; (d) number and proportion of tweets labeled as negative or positive per sector

Because tweets for the year 2022 were collected only for January, that year is excluded from Figs. 4a and 4b, but otherwise those tweets are included in the statistics. Fig. 4a shows the total number of tweets collected per year. As can be seen, the majority of collected tweets were from 2018–2021 because TWINT had limited ability to collect older tweets and the hospitals created their Twitter accounts only recently; moreover, the COVID-19 pandemic might have played a role in increasing the number of tweets, especially during and after 2020. Fig. 4b shows that negative tweets outnumbered positive tweets in all years except for 2012. This is reasonable because the dataset has a larger proportion of negative tweets. Moreover, the authors noticed that a substantial amount (ca. 80%) of tweets published in 2016 and 2017 were negative and that 2012 had the smallest proportion of negative tweets (ca. 30%). In January 2022, 423 tweets were collected, of which 106 (25%) were positive and 317 (75%) were negative.

After annotating tweets based on the referenced sector, the results show that the dataset has a fair proportion of tweets relating to the public (53.8%) and private (44.9%) sectors. During the annotation, the sector could not be specified for ca. 160 tweets (see Fig. 4c). The resulting counts of positive and negative tweets for each sector are summarized in Fig. 4d. As can be seen, a huge fraction (71.5%) of the private-sector tweets were negative, whereas the public sector had a more balanced distribution with 43% positive and 57% negative. The public sector had the largest proportion of positive tweets, i.e., the public sector contributed more significantly to positive tweets.

4.2 Support Vector Machine Results

The results obtained after training the four models with 10-fold cross-validation are summarized in Table 3. Each row in the table corresponds to a combination of an SVM kernel and an AraVec architecture; for example, the first row (Linear SVM/CBoW) shows the results of using a linear SVM kernel and the CBoW AraVec architecture. The average (*Avg*), minimum (*Min*), maximum (*Max*), and standard deviation (*SD*) of each experiment are also reported in Table 3 in terms of the three chosen evaluation measures. The highest value obtained among the different configurations is highlighted in bold. Each experiment in this section took ca. 15 min to complete for an input matrix of size $12,400 \times 300$ (12,400 tweets each, represented by 300 features).

Table 3: Results of experimenting with different kernels and AraVec architectures for the SVM model, reported in percentages

Configuration	*Accuracy*				*Micro-F*1				*Macro-F*1			
	Avg	Min	Max	SD	Avg	Min	Max	SD	Avg	Min	Max	SD
Linear SVM/CBoW	95.71	94.84	96.69	0.6	95.71	94.84	96.69	0.6	95.37	94.50	96.39	0.6
Linear SVM/skip-gram	**96.30**	**95.32**	96.94	0.5	**96.30**	**95.32**	96.94	0.5	**96.00**	**95.00**	96.68	0.5
RBF SVM/CBoW	95.46	89.11	**97.98**	2.4	95.46	89.11	**97.98**	2.4	95.05	87.78	97.83	2.7
RBF SVM/skip-gram	95.48	89.27	97.58	2.4	95.48	89.27	97.58	2.4	95.07	87.86	97.39	2.7

Based on these results, the best-performing model in terms of all three evaluation measures was the SVM with a linear kernel and the skip-gram architecture for AraVec, with an average *Accuracy* and *Micro-F*1 of 96.3%, an average *Macro-F*1 of 96.0%, and the lowest standard deviation. Table 3 also shows that skip-gram outperformed CBoW regardless of the kernel used, which supports the argument in Al-Twairesh [56] that skip-gram is superior.

4.3 Tuning MARBERT-Based Models

4.3.1 Tuning MARBERT–BiLSTM Model

For the MARBERT–BiLSTM model, the authors experimented with 13 different configurations (CONF1–CONF13) by changing the size of the hidden layer in BiLSTM, the batch size, the number of epochs, and the learning rate; they used the recommended values of the number of epochs, batch size, and Adam learning rate [22]. Moreover, the authors experimented with adding dropout and discarding the text normalization steps. Table 4 gives the experimental hyperparameters used for the different configurations. The results obtained by the different MARBERT–BiLSTM configurations when trained on HoPE-SA are reported in Table 5, showing the best scores in bold.

Table 4: MARBERT–BiLSTM hyperparameters used in 13 configurations in hyperparameter tuning experiments conducted on the HoPE-SA dataset

Configuration	Normalization	Learning rate	Hidden layer size	Batch size	Epochs	Drop-out
CONF1	Yes	5×10^{-5}	50	32	2	—
CONF2	No	5×10^{-5}	50	32	2	—
CONF3	No	5×10^{-5}	768	32	2	—
CONF4	No	5×10^{-5}	50	16	2	—
CONF5	No	5×10^{-5}	50	32	4	—
CONF6	No	5×10^{-5}	50	32	2	0.5
CONF7	No	2×10^{-5}	50	32	2	—
CONF8	No	2×10^{-5}	50	32	4	—
CONF9	No	2×10^{-5}	50	32	2	0.5
CONF10	No	2×10^{-5}	50	16	4	—
CONF11	No	2×10^{-5}	768	32	2	—
CONF12	No	2×10^{-5}	768	32	4	—
CONF13	Yes	2×10^{-5}	50	16	4	—

Table 5: Results of experiments on tuning MARBERT–BiLSTM hyperparameters, reported in percentages

Configuration	*Accuracy*				*Micro-F*1				*Macro-F*1			
	Avg	Min	Max	SD	Avg	Min	Max	SD	Avg	Min	Max	SD
CONF1	98.28	97.76	98.72	0.35	98.32	97.78	98.71	0.34	98.19	97.61	98.62	0.37
CONF2	98.30	97.69	98.80	0.34	98.34	97.99	98.99	0.32	98.21	97.83	98.71	0.35
CONF3	98.35	98.16	98.68	0.20	98.38	98.18	98.79	0.25	98.26	98.04	98.69	0.26
CONF4	98.22	97.74	98.55	0.33	98.22	97.74	98.55	0.33	98.10	97.54	98.45	0.37
CONF5	98.19	97.44	98.60	0.47	98.24	97.46	98.59	0.47	98.10	97.27	98.50	0.51
CONF6	98.38	98.04	98.64	0.22	98.39	98.06	98.71	0.23	98.23	97.91	98.6	0.24
CONF7	98.55	98.32	98.88	0.27	98.57	98.34	98.87	0.25	98.46	98.21	98.79	0.26
CONF8	**98.74**	**98.48**	98.88	0.16	**98.75**	98.51	98.87	0.15	**98.65**	**98.39**	98.80	0.16
CONF9	98.65	98.20	98.68	0.24	98.48	98.23	98.71	0.22	98.36	98.09	98.62	0.24
CONF10	**98.74**	98.39	**99.11**	0.28	98.74	**99.39**	**99.11**	0.28	**98.65**	98.26	**99.03**	0.30
CONF11	98.65	98.20	98.96	0.32	98.66	98.23	98.95	0.28	98.58	98.09	98.88	0.31
CONF12	98.70	98.28	99.00	0.30	98.71	98.31	98.99	0.30	98.62	98.17	98.93	0.32
CONF13	98.62	98.31	98.79	0.18	98.62	98.31	98.79	0.18	98.51	98.17	98.71	0.20

Table 5 shows slight variations among the experiments. Nevertheless, it is notable that removing the text normalization step improved the model performance, albeit by a small margin. Furthermore, changing the learning rate from 5×10^{-5} to 2×10^{-5} yielded a relatively large improvement. Meanwhile, CONF8 and CONF10 had similar performances when considering the average of the measures, but CONF8 had the lowest variation in performance. Thus, the best combination of hyperparameters that the authors found was that in CONF8. Furthermore, the worst combinations of hyperparameters were found in CONF4 and CONF5.

4.3.2 Tuning MARBERT–FNN Model

To optimize the MARBERT–FNN model, similar experiments were conducted but with the hyperparameters related to the BiLSTM layer excluded, and the authors tested the recommended values of the number of epochs, batch size, and Adam learning rate [22]. Because MARBERT was pre-trained on minimally processed Arabic tweets, the authors tested its performance with and without the additional steps of normalization. Table 6 summarizes the hyperparameters used in the different configurations for the experiments. Notably, CONF14 used the default settings for the hyperparameters.

Table 6: MARBERT–FNN hyperparameters used in eight configurations in hyperparameter tuning experiments conducted on the HoPE-SA dataset

Configuration	Normalization	Learning rate	Hidden layer size	Batch size	Epochs	Drop-out
CONF14	Yes	5×10^{-5}	50	32	2	—
CONF15	Yes	5×10^{-5}	50	32	2	0.5
CONF16	Yes	5×10^{-5}	50	32	4	—
CONF17	Yes	5×10^{-5}	50	16	2	—
CONF18	No	5×10^{-5}	50	32	2	—
CONF19	Yes	2×10^{-5}	50	32	2	—
CONF20	Yes	2×10^{-5}	100	32	2	—
CONF21	Yes	2×10^{-5}	50	32	2	0.5

Table 7 reports the results obtained by the different MARBERT–FNN configurations when trained over the HoPE-SA dataset, highlighting the best scores in bold. Changing the default values of the number of epochs and the batch size as well as eliminating normalization resulted in lower performance in all measures compared to the default settings but setting the Adam learning rate to 2×10^{-5} enhanced the performance of the model. Therefore, the authors fixed the learning rate to 2×10^{-5} while changing the size of the FNN layer and adding a dropout layer of 0.5 in separate experiments—CONF20 and CONF21—because the latter hyperparameters gave the best results, second to those obtained by changing the learning rate. These combinations did not exceed the improvement created by the learning rate, which is represented by the CONF19 model in Table 6.

Table 7: Results of experiments on tuning MARBERT–FNN hyperparameters, reported in percentages

Configuration	*Accuracy*				*Micro-F*1				*Macro-F*1			
	Avg	Min	Max	SD	Avg	Min	Max	SD	Avg	Min	Max	SD
CONF14	98.53	98.20	98.84	0.25	98.54	98.19	98.83	0.26	98.43	98.02	98.73	0.28
CONF15	98.49	98.24	98.84	0.27	98.51	98.27	98.51	0.27	98.40	98.13	98.73	0.26
CONF16	98.31	97.96	98.72	0.33	98.33	98.02	98.71	0.32	98.20	97.87	98.61	0.34
CONF17	98.25	98.06	98.47	0.18	98.43	97.96	98.47	0.18	98.11	97.93	98.34	0.18
CONF18	98.43	97.98	98.79	0.31	98.43	97.98	98.79	0.29	98.31	97.82	98.70	0.32
CONF19	**98.65**	98.12	**99.32**	0.49	**98.68**	98.19	**99.31**	0.47	**98.58**	98.06	**99.27**	0.51
CONF20	98.61	**98.52**	98.76	0.09	98.63	**98.55**	98.75	0.09	98.52	**98.42**	98.66	0.09
CONF21	98.63	98.20	99.04	0.38	98.65	98.22	99.03	0.36	98.54	98.08	98.97	0.39

4.4 BERT-Based Model Results

This subsection provides the results of the 10-fold cross-validation based on the best combinations of hyperparameters that were found previously. For MARBERT–BiLSTM, the authors used the same hyperparameters in CONF8 as those given in Table 4. Moreover, for comparison, the authors used the same settings of fine-tuned hyperparameters for mBERT–BiLSTM. Additionally, the authors used the hyperparameter combination from CONF19 for the MARBERT–FNN and mBERT–FNN fine-tuned models. After evaluation, the MARBERT–BiLSTM model had the highest performance, while the MARBERT–FNN model had a slightly lower performance. Conversely, the mBERT–FNN model outperformed the mBERT–BiLSTM model, but again by a small margin. Overall, there was a notable difference in the performance of the MARBERT-based models and the mBERT-based models. Table 8 summarizes the results of the authors' final BERT-based models, showing the highest scores in bold. The times consumed by the BiLSTM variants of MARBERT and mBERT were 4 h 6 m 40 s and 4 h 4 m 13 s, respectively. In contrast, the times consumed by the FNN variants of MARBERT and BiLSTM were 1 h 28 m 42 s and 1 h 28 m 34 s, respectively. Fig. 5 displays the confusion matrices for the authors' models.

Table 8: Results of experiments on tuning MARBERT–FNN hyperparameters, reported in percentages

Configuration	*Accuracy*				*Micro-F*1				*Macro-F*1			
	Avg	Min	Max	SD	Avg	Min	Max	SD	Avg	Min	Max	SD
MARBERT–BiLSTM	**98.71**	**98.18**	**99.25**	0.39	**98.73**	**98.22**	**99.27**	0.36	**98.63**	**98.09**	**99.21**	0.40
mBERT–BiLSTM	96.10	94.07	97.04	0.88	96.11	94.15	97.02	0.86	95.81	93.72	96.87	0.93
MARBERT–FNN	98.51	96.84	98.88	0.33	98.52	97.91	98.87	0.31	98.41	97.75	98.80	0.34
mBERT–FNN	96.12	95.56	96.74	0.44	96.15	95.50	96.77	0.43	95.86	95.25	96.51	0.56

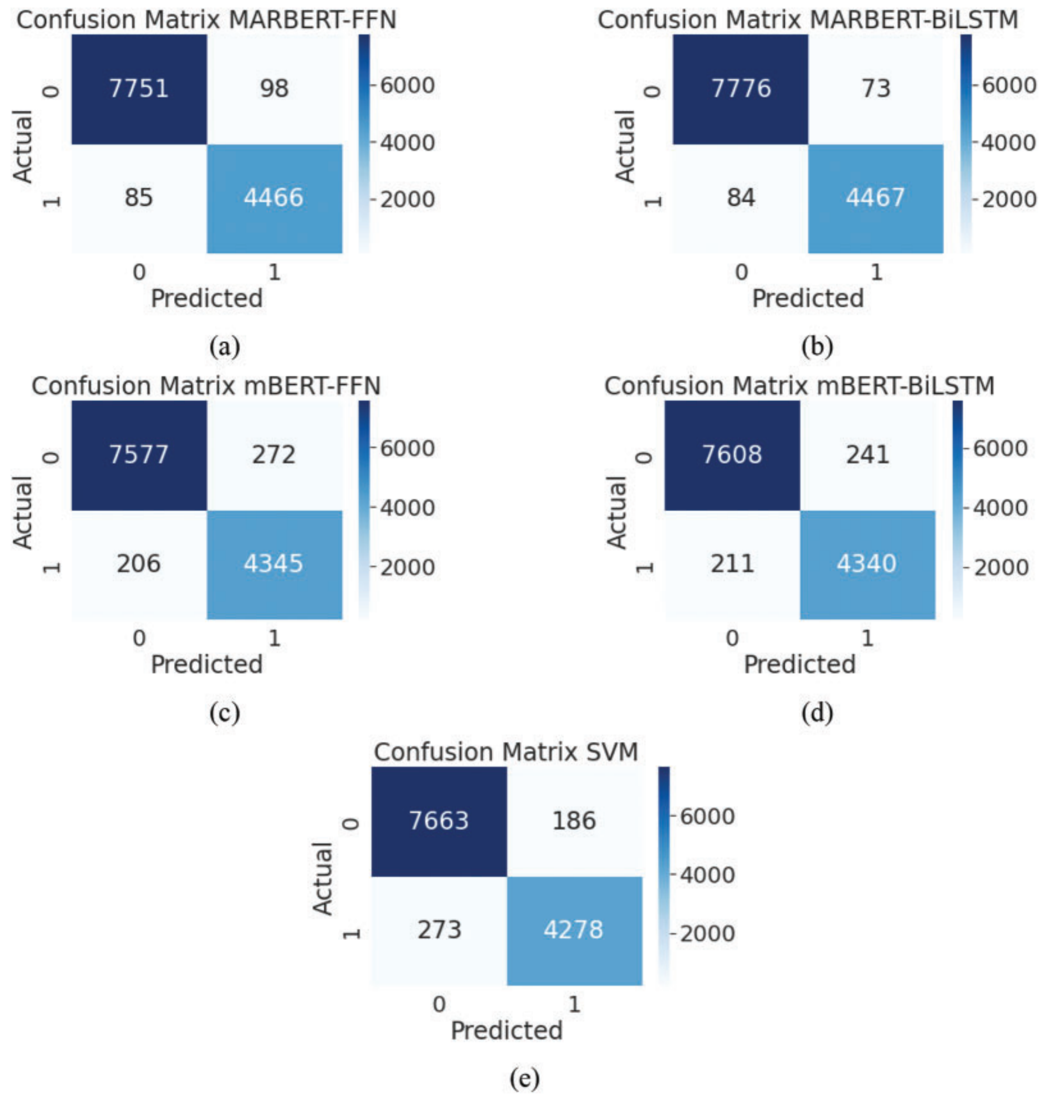

Figure 5: Confusion matrices for final models: (a) MARBERT–FNN 10-fold; (b) MARBERT–BiLSTM 10-fold; (c) mBERT–FNN; (d) mBERT–BiLSTM; (e) linear-kernel SVM with skip-gram

5 Discussion

A primary contribution of this work is the introduction of HoPE-SA, a patient experience corpus that explores carefully curated data related to patient experiences written in Arabic on Twitter and within the region of Saudi Arabia. The authors attempted to ensure that tweets were retrieved as objectively as possible.

The first objective was to explore the attitudes of patients in Saudi Arabia toward the provided healthcare services. From the statistical analysis applied to HoPE-SA to examine the patients' sentiments about the healthcare services provided in Saudi Arabia, the authors generally conclude that patients in Saudi Arabia are not fully satisfied with the healthcare services provided because 63.3% of the tweets from the collected dataset were negative.

The second objective was to determine whether a tangible difference existed between the quality of services provided by private and public healthcare sectors from the perspective of their patients. The statistical analysis showed that the patients of the private sector seem to be less satisfied than the patients of the public sector. This could be because public healthcare is free, unlike private healthcare whose patients may expect a better service. By exploring the dataset, the authors found that many complaints in the public sector were about delays in appointments. However, tweets about the private sector compared waiting times to those in the public sector and expressed discontent due to having similar waiting times despite paying for healthcare.

The third objective was to examine the performance impact of extracting more features during the fine-tuning step of MARBERT using a BiLSTM layer. The experimental study showed that the margin of enhancement gained in the MARBERT-based models by adding a BiLSTM layer was small (only two in a thousand tweets), but the time difference was more than double. This time-to-performance trade-off might not be satisfactory in most domains. To obtain another perspective on the performance of their models, the authors trained their optimized models on AraSarcasm-v2 [11] and compared the results with those of the SA task. The best result was achieved by El Mahdaouy et al. [66] using a MARBERT-based model with a macro-F1 score of 0.6625. Using the same training and testing set, the authors obtained macro-F1 scores of 0.6608 and 0.6471 for MARBERT–FNN and MARBERT–BiLSTM, respectively. The authors' MARBERT–FNN tuning has proven to be more universal than their MARBERT–BiLSTM model, achieving the highest macro-F1 score after that of El Mahdaouy et al. [66]. However, when evaluating it on the authors' dataset, the fine-tuning by MARBERT–BiLSTM had higher scores than the fine-tuning by MARBERT–FNN. This could indicate that BiLSTM was more sensitive to the hyperparameters and dataset domain. Moreover, using the authors' best-performing SVM model (linear kernel and skip-gram AraVec) on the ArSarcasm-v2 dataset resulted in an average macro-F1 score of 0.6001. The small differences in the BERT-like models' performances on ArSarcasm-v2 compared with the best model's performance could be attributed to the fact that the authors obtained their final models' settings by fine-tuning them on their self-collected dataset, whereas the highest scores achieved by El Mahdaouy et al. [66] were achieved by tuning their model on the AraSarcasm-v2 dataset. Furthermore, there was a substantial difference between AraSarcasm-v2 and the authors' dataset. First, the authors' dataset is domain-specific and only targets patient experience tweets, whereas ArSarcasm-v2 is more diverse and covers several issues. Moreover, the authors' tuning was carried out to enhance the classification of two classes only (positive and negative). However, to compare the present results with those of El Mahdaouy et al. [66], the authors changed the output of their models into three classes (positive, neutral, and negative).

Regarding preprocessing steps, the SVM model required extensive preprocessing to improve its performance; this may have been because AraVec was trained on preprocessed and normalized text. Furthermore, in MARBERT, extensive preprocessing did not necessarily improve the performance. As can be seen from Table 5, removing normalization had only a slight effect on the models' performances and even enhanced the performances for some models. The authors attributed this to the fact that MARBERT was originally trained on a dataset of minimally preprocessed tweets, and thus the model was not disturbed by noise and could potentially extract the semantics behind the tweets. Moreover, the WordPiece tokenizer that was utilized by the BERT-based models was rarely hindered by the out-of-dictionary problem because it broke words into pieces. This implies that BERT, with its Word-Piece tokenization, might deem extensive text processing futile in the same way that DL and prediction-based embeddings may deem feature extraction and lexicons ineffective.

Finally, when comparing all five models, the authors found that the MARBERT models had superior performances. The authors observed notable enhancements in the models that utilized

MARBERT rather than mBERT, although they had the same architecture and settings. Moreover, the mBERT and the SVM AraVec baseline models had similar performances, although the pre-training of AraVec was performed on a dialectic Arabic corpus (similar to MARBERT) while mBERT was pre-trained on a corpus that contained mainly MSA. From this, the authors concluded that not only did the dialectic pre-training corpus contribute to their MARBERT models' good performances but also that the transformer architecture made a considerable contribution to its superiority because mBERT was able to achieve a relatively high performance despite the nature of its pre-training data.

6 Conclusions

SA has been studied extensively as a tool that can automate the process of surveying and analyzing the opinions of people. This study aimed to bridge the gap in medical-related applications of SA and to introduce an Arabic patient experience corpus. The newly constructed open-source HoPE-SA dataset is intended to possibly bridge the gap in Arabic NLP resources and corpora, especially in the health sector. The findings in this study can inspire additional investigations and aid the comprehension of the Saudi culture of health, as provided by real-time Twitter data.

The authors' experiments resulted in five models. The first two were MARBERT models, one fine-tuned with a BiLSTM and the other with an FNN. Similarly, the authors used two mBERT models fine-tuned with a BiLSTM and an FNN, and finally an SVM AraVec model. For evaluation, the authors used 10-fold cross-validation and measured the performance using the metrics of accuracy and F1 score. After evaluation, the authors concluded the following: the MARBERT-based models yielded superior performances, especially the BiLSTM fine-tuned model. The shortcomings of mBERT were due to the formal Arabic on which it was trained; the classic SVM AraVec model had slightly better but comparable performance to the mBERT models despite its simpler architecture, which may have been due to AraVec's pre-training dataset, which contained mostly DA as well as MSA.

In the future, the authors intend to conduct a comparative study assessing the generalizability of the MARBERT–BiLSTM model when applied to a benchmark of several datasets and compare its performance to that of state-of-the-art methods in terms of efficiency and effectiveness. In addition, the authors plan to implement an aspect-based SA approach to identify the specific services (aspects) alluded to in each tweet and assign a sentiment label to each service, which will help healthcare organizations identify which services need the most improvement. Finally, the authors also plan to extend their study from binary SA to multi-class emotion analysis.

Acknowledgement: The authors thank the anonymous reviewers for their constructive comments.

Funding Statement: The authors received no specific funding for this study.

Availability of Data and Materials: The newly constructed HoPE-SA dataset is publicly available at https://github.com/HoPESA-KSU/Patient-Experience-SA (accessed on 30 January 2023).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] F. Ahmed, J. Burt and M. Roland, "Measuring patient experience: Concepts and methods, The Patient," *Patient-Centered Outcomes Research*, vol. 7, no. 3, pp. 235–241, 2014.

[2] W. Wiersma, "The validity of surveys: Online and offline," *Oxford Internet Institute*, vol. 18, no. 3, pp. 321–340, 2013.

[3] M. Karatas, L. Eriskin, M. Deveci, D. Pamucar and H. Garg, "Big data for healthcare industry 4.0: Applications, challenges and future perspectives," *Expert Systems with Applications*, vol. 200, pp. 116912, 2022.

[4] N. Radwan, "The Internet's role in undermining the credibility of the healthcare industry," *International Journal of Computations, Information and Manufacturing (IJCIM)*, vol. 2, no. 12022. https://doi.org/10.54489/ijcim.v2i1.74

[5] R. van Kessel, B. L. Wong, T. Clemens and H. Brand, "Digital health literacy as a super determinant of health: More than simply the sum of its parts," *Internet Interventions*, vol. 27, 2022. https://doi.org/10.1016/j.invent.2022.100500

[6] H. K. Aldayel and A. M. Azmi, "Arabic tweets sentiment analysis—A hybrid scheme," *Journal of Information Science*, vol. 42, no. 6, pp. 782–797, 2016.

[7] Similarweb, "Top Websites ranking in Saudi Arabia in September 2021," 2021. [Online]. Available: https://www.similarweb.com/top-websites/saudi-arabia/

[8] A. Bayazed, O. Torabah, R. AlSulami, D. Alahmadi, A. Babour *et al.,* "SDCT: Multi-dialects corpus classification for Saudi tweets," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, pp. 216–223, 2020.

[9] M. Nabil, M. Aly and A. Atiya, "ASTD: Arabic sentiment tweets dataset," in *2015 Conf. on Empirical Methods in Natural Language Processing (EMNLP'15)*, Lisbon, Portugal, pp. 2515–2519, 2015.

[10] A. El Mahdaouy, A. El Mekki, K. Essefar, N. El Mamoun, I. Berrada *et al.,* "ArSAS: An Arabic speech-act and sentiment corpus of tweets," in *3rd Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT'18)*, Miyazaki, Japan, pp. 20–25, 2018.

[11] I. Abu Farha and W. Magdy, "From Arabic sentiment analysis to sarcasm detection: The ArSarcasm dataset," in *4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT4), with a Shared Task on Offensive Language Detection*, Marseille, France, pp. 32–39, 2020.

[12] V. S. Pagolu, K. N. Reddy, G. Panda and B. Majhi, "Sentiment analysis of Twitter data for predicting stock market movements," in *2016 Int. Conf. on Signal Processing, Communication, Power and Embedded System (SCOPES'16)*, Paralakhemundi, India, pp. 1345–1350, 2016.

[13] T. Elghazaly, A. Mahmoud and H. A. Hefny, "Political sentiment analysis using Twitter data," in *2nd Int. Conf. on Internet of Things and Cloud Computing (ICC'16)*, Cambridge, UK, pp. 1–5, 2016.

[14] M. Hadwan, M. Al-Hagery, M. Al-Sarem and F. Saeed, "Arabic sentiment analysis of users' opinions of governmental mobile applications," *Computers, Materials and Continua*, vol. 72, no. 3, pp. 4675–4689, 2022.

[15] J. R. Saura, P. Palos-Sanchez and A. Grilo, "Detecting indicators for startup business success: Sentiment analysis using text data mining," *Sustainability*, vol. 11, no. 3, pp. 917, 2019.

[16] P. Baid, A. Gupta and N. Chaplot, "Sentiment analysis of movie reviews using machine learning techniques," *International Journal of Computer Applications*, vol. 179, no. 7, pp. 45–49, 2017.

[17] H. Rahman, J. Tariq, M. A. Masood, A. F. Subahi, O. I. Khalaf *et al.,* "Multi-tier sentiment analysis of social media text using supervised machine learning," *Computers, Materials and Continua*, vol. 74, no. 3, pp. 5527–5543, 2023.

[18] G. Badaro, R. Baly, H. Hajj, W. El-Hajj, K. Shaban *et al.,* "A survey of opinion mining in Arabic: A comprehensive system perspective covering challenges and advances in tools, resources, models, applications, and visualizations," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, no. 3, pp. 1–52, 2019.

[19] T. Young, D. Hazarika, S. Poria and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," in *31st Conf. on Neural Information Processing Systems (NIPS'17)*, Long Beach, CA, USA, pp. 1–11, 2017.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *17th Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT'19)*, Minneapolis, MN, USA, pp. 4171–4186, 2018.

[23] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2nd ed., Upper Saddle River, NJ, USA: Prentice Hall, 2009.

[24] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "*Learning Internal Representations by Error Propagation*, Technical Report No. ICS–8506. La Jolla Institute for Cognitive Science, University of California, San Diego, CA, USA," 1985.

[25] M. Abdul-Mageed, A. Elmadany and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep bidirectional transformers for Arabic," in *59th Annual Meeting of the Association for Computational Linguistics and 11th International Joint Conf. on Natural Language Processing (ACL-IJCNLP'21)*, Stroudsburg, PA, USA, pp. 7088–7105, 2021.

[26] I. Abu Farha and W. Magdy, "A comparative study of effective approaches for Arabic sentiment analysis," *Information Processing and Management*, vol. 58, no. 2, pp. 102438, 2021.

[27] Q. T. Nguyen, T. L. Nguyen, N. H. Luong and Q. H. Ngo, "Fine-tuning BERT for sentiment analysis of Vietnamese reviews," in *7th NAFOSTED Conf. on Information and Computer Science (NICS'20)*, Ho Chi Minh City, Vietnam, pp. 302–307, 2020.

[28] F. Souza, R. Nogueira and R. Lotufo, "Portuguese named entity recognition using BERT-CRF," *CoRR*, vol. abs/1909.10649, pp. 1–8, 2019. https://doi.org/10.48550/arXiv.1909.10649

[29] P. M. Nadkarni, L. Ohno-Machado and W. W. Chapman, "Natural language processing: An introduction," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, 2011.

[30] J. Ramos, "Using TF-IDF to determine word relevance in document queries," in *1st Instructional Conf. on Machine Learning (ICML'03)*, Washington, DC, USA, pp. 29–48, 2003.

[31] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," in *1st Int. Conf. on Learning Representations (ICLR'13)*, Scottsdale, AZ, USA, 2013.

[32] D. Suleiman and A. Awajan, "Comparative study of word embeddings models and their usage in Arabic language applications," in *2018 Int. Arab Conf. on Information Technology (ACIT'18)*, Werdanye, Lebanon, pp. 1–7, 2018.

[33] A. Mahdaouy, E. Gaussier and S. Ouatik El Alaoui, "Arabic text classification based on word and document embeddings," in *2nd Int. Conf. on Advanced Intelligent Systems and Informatics (AISI'16)*, Cairo, Egypt, pp. 32–41, 2016.

[34] K. S. Kalaivani, S. Uma and C. S. Kanimozhiselvi, "Comparison of deep learning approaches for sentiment classification," in *6th Int. Conf. on Inventive Computation Technologies (ICICT'21)*, Coimbatore, India, pp. 1043–1047, 2021.

[35] T. K. Tran, H. M. Dinh and T. T. Phan, "Building an enhanced sentiment classification framework based on natural language processing," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 43, no. 21, pp. 1771–1777, 2022.

[36] E. Cambria, D. Das, S. Bandyopadhyay and A. Feraco, Affective computing and sentiment analysis. In: *A Practical Guide to Sentiment Analysis*. Vol. 5. Berlin/Heidelberg: Springer, pp. 1–10, 2017.

[37] K. C. Sewalk, G. Tuli, Y. Hswen, J. S. Brownstein and J. B. Hawkins, "Using Twitter to examine web-based patient experience sentiments in the United States: Longitudinal study," *Journal of Medical Internet Research*, vol. 20, no. 10, pp. e10043, 2018.

[38] S. Liu, J. Li and J. Liu, "Leveraging transfer learning to analyze opinions, attitudes, and behavioral intentions toward COVID-19 vaccines: Social media content and temporal analysis," *Journal of Medical Internet Research*, vol. 23, no. 8, pp. e30251, 2021.

[39] A. M. Alayba, V. Palade, M. England and R. Iqbal, "Arabic language sentiment analysis on health services," in *1st Int. Workshop on Arabic Script Analysis and Recognition (ASAR'17)*, Nancy, France, pp. 114–118, 2017.

[40] S. S. Aljameel, D. A. Alabbad, N. A. Alzahrani, S. M. Alqarni, F. A. Alamoudi *et al.,* "A sentiment analysis approach to predict an individual's awareness of the precautionary procedures to prevent COVID-19 outbreaks in Saudi Arabia," *International Journal of Environmental Research and Public Health*, vol. 18, no. 1, pp. 218, 2021.
[41] R. M. Alahmary, H. Z. Al-Dossari and A. Z. Emam, "Sentiment analysis of Saudi dialect using deep learning techniques," in *2019 Int. Conf. on Electronics, Information, and Communication (ICEIC'19)*, Auckland, New Zealand, pp. 1–6, 2019.
[42] A. Dahou, S. Xiong, J. Zhou, M. H. Haddoud and P. Duan, "Word embeddings and convolutional neural network for Arabic sentiment classification," in *26th Int. Conf. on Computational Linguistics (COLING'16)*, Osaka, Japan, pp. 2418–2427, 2016.
[43] M. Heikal, M. Torki and N. El-Makky, "Sentiment analysis of Arabic tweets using deep learning," *Procedia Computer Science*, vol. 142, pp. 114–122, 2018.
[44] A. B. 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.
[45] A. Abdelali, S. Hassan, H. Mubarak, K. Darwish and Y. Samih, "Pre-training BERT on Arabic tweets: Practical considerations," *CoRR*, vol. abs/2102.10684, pp. 1–6, 2021. https://doi.org/10.48550/arXiv.2102.10684
[46] L. Bashmal and D. AlZeer, "ArSarcasm shared task: An ensemble BERT model for sarcasm detection in Arabic tweets," in *6th Arabic Natural Language Processing Workshop (WANLP'21)*, Kyiv, Ukraine, pp. 323–328, 2021.
[47] H. Chouikhi, H. Chniter and F. Jarray, "Arabic sentiment analysis using BERT model," in *2021 Int. Conf. on Computational Collective Intelligence (ICCCI'21)*, Rhodes, Greece, pp. 621–632, 2021.
[48] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek *et al.,* "Unsupervised cross-lingual representation learning at scale," in *58th Annual Meeting of the Association for Computational Linguistics (ACL'20)*, Stroudsburg, PA, USA, pp. 8440–8451, 2020.
[49] I. Abu Farha and W. Magdy, "Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection," in *6th Arabic Natural Language Processing Workshop (WANLP'21)*, Kyiv, Ukraine, pp. 21–31, 2021.
[50] W. Antoun, F. Baly and H. Hajj, "AraBERT: Transformer-based model for Arabic language understanding," in *4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT4), with a Shared Task on Offensive Language Detection*, Marseille, France, pp. 9–15, 2020.
[51] M. Naski, A. Messaoudi, H. Haddad, M. BenHajhmida, C. Fourati *et al.,* "iCompass at shared task on sarcasm and sentiment detection in Arabic," in *6th Arabic Natural Language Processing Workshop (WANLP'21)*, Kyiv, Ukraine, pp. 381–385, 2021.
[52] Y. Kim, "Convolutional neural networks for sentence classification," in *2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP'14)*, Lisbon, Portugal, pp. 1746–1751, 2014.
[53] X. Zhang, J. Zhao and Y. LeCun, "Character-level convolutional networks for text classification," in *28th Int. Conf. on Neural Information Processing Systems (NIPS'15)*, Montreal, Canada, pp. 1–9, 2015.
[54] S. Lyu and J. Liu, "Convolutional recurrent neural networks for text classification," *Journal of Database Management*, vol. 32, no. 4, pp. 65–82, 2021.
[55] J. D. Lafferty, A. McCallum and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *8th Int. Conf. on Machine Learning (ICML'01)*, San Francisco, CA, USA, pp. 282–289, 2001.
[56] N. Al-Twairesh, "The evolution of language models applied to emotion analysis of Arabic tweets," *Information*, vol. 12, no. 2, pp. 84, 2021.
[57] C. Zacharias, "TWINT project: Open-source Twitter intelligence," 2021. [Online]. Available: https://github.com/twintproject/twint
[58] Cybermetrics Lab, Consejo Superior de Investigaciones Científicas (CSIC), "Ranking web of hospitals—Saudi Arabia," 2021. [Online]. Available: https://hospitals.webometrics.info/en/aw/saudi%20arabia%20

[59] Saudi Central Board for Accreditation of Healthcare Institutions (CBAHI), "Hospitals accreditation status till 4 Jan 2021," 2021. [Online]. Available: https://portal.cbahi.gov.sa/Library/Assets/eng-hospital-123206.pdf
[60] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, no. 5, pp. 378–382, 1971.
[61] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.
[62] S. A. A. Hakami, R. Hendley and P. Smith, "Arabic emoji sentiment lexicon (Arab-ESL): A comparison between Arabic and European emoji sentiment lexicons," in *6th Arabic Natural Language Processing Workshop (WANLP'21)*, Kyiv, Ukraine, pp. 60–71, 2021.
[63] Google, "Welcome to colaboratory," 2022. [Online]. Available: https://colab.research.google.com/
[64] C. W. Hsu, C. C. Chang and C. J. Lin, "*A practical guide to support vector classification*, Technical Report. Department of Computer Science, National Taiwan University, Taipei, Taiwan," 2003. [Online]. Available: https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf
[65] E. A. Khalil, E. M. F. El Houby and H. K. Mohamed, "Deep learning for emotion analysis in Arabic tweets," *Journal of Big Data*, vol. 8, no. 1, pp. 136, 2021.
[66] A. El Mahdaouy, A. El Mekki, K. Essefar, N. El Mamoun, I. Berrada *et al.,* "Deep multi-task model for sarcasm detection and sentiment analysis in Arabic language," in *6th Arabic Natural Language Processing Workshop (WANLP'21)*, Kyiv, Ukraine, pp. 334–339, 2021.

Supplementary Material

Supplementary Table 1: Alphabetical list of hospitals included in HoPE-SA dataset, along with the sector (public or private) of each according to CBAHI

Seq.	Hospital	Sector
1	Al Hammadi Hospital	Private
2	Al Mowasat Hospital	Private
3	Almana Hospital	Private
4	Dallah Hospital	Private
5	Dr. Erfan and Bagedo General Hospital—Jeddah	Private
6	Dr. Sulaiman Al Habib Medical Group	Private
7	King Abdulaziz Medical City—Riyadh	Public
8	King Fahad Specialist Hospital—Buraydah	Public
9	King Fahd Hospital of University Education—Al Khobar	Public
10	King Fahd Specialist Hospital—Dammam	Public
11	King Faisal Specialist Hospital and Research Centre— Riyadh	Public
12	King Khalid Eye Specialist Hospital	Public
13	King Saud Medical City—Riyadh	Public
14	Maghrabi Hospital	Private