Received: 5 October 2023 | Revised: 12 February 2024 | Accepted: 19 February 2024

DOI: 10.1002/eng2.12874

RESEARCH ARTICLE

Engineering Reports Open Access WILEY

A hybrid neural network model based on transfer learning for Arabic sentiment analysis of customer satisfaction

Duha Mohamed Adam Bakhit[1] | **Lawrence Nderu**[2] | **Antony Ngunyi**[3]

[1]Department of Mathematics, Pan African University, Institute for Basic Sciences, Technology and Innovation, Nairobi, Kenya

[2]School of Computing and Information Technology, Department of Computing, Jomo Kenyatta University of Agriculture and Technology, Nairobi, Kenya

[3]Statistics and Actuarial Sciences Department, Dedan Kimathi University of Technology, Nyeri, Kenya

Correspondence
Duha Mohamed Adam Bakhit, Department of Mathematics, Pan African University, Institute for Basic Sciences, Technology and Innovation, Nairobi, Kenya.
Email: bakhit.duha@students.jkuat.ac.ke

Abstract

Sentiment analysis, a method used to classify textual content into positive, negative, or neutral sentiments, is commonly applied to data from social media platforms. Arabic, an official language of the United Nations, presents unique challenges for sentiment analysis due to its complex morphology and dialectal diversity. Compared to English, research on Arabic sentiment analysis is relatively scarce. Transfer learning, which applies the knowledge learned from one domain to another, can address the limitations of training time and computational resources. However, the development of transfer learning for Arabic sentiment analysis is still underdeveloped. In this study, we develop a new hybrid model, RNN-BiLSTM, which merges recurrent neural networks (RNN) and bidirectional long short-term memory (BiLSTM) networks. We used Arabic bidirectional encoder representations from transformers (AraBERT), a state-of-the-art Arabic language pre-trained transformer-based model, to generate word-embedding vectors. The RNN-BiLSTM model integrates the strengths of RNN and BiLSTM, including the ability to learn sequential dependencies and bidirectional context. We trained the RNN-BiLSTM model on the source domain, specifically the Arabic reviews dataset (ARD). The RNN-BiLSTM model outperforms the RNN and BiLSTM models with default parameters, achieving an accuracy of 95.75%. We further applied transfer learning to the RNN-BiLSTM model by fine-tuning its parameters using random search. We compared the performance of the fine-tuned RNN-BiLSTM model with the RNN and BiLSTM models on two target domain datasets: ASTD and Aracust. The results showed that the fine-tuned RNN-BiLSTM model is more effective for transfer learning, achieving an accuracy of 95.44% and 96.19% on the ASTD and Aracust datasets, respectively.

KEYWORDS

AraBERT, Arabic sentiment analysis, neural network, transfer learning

1 | INTRODUCTION

As the Internet becomes more widely used, social media networks have become essential for sharing information and connecting with people around the world.[1–3] Many users use blogs and social media to express their opinions on individuals, events, places, and products,[4,5] impacting many businesses through their comments. Sentiment analysis (SA) is a research field that aims to identify the emotions and opinions of a specific group toward a particular topic or product.[6] It employs natural language processing (NLP), computational linguistics, text analysis, and machine learning (ML) to identify the sentiment of a phrase, classifying it as positive, negative, or neutral. SA has numerous analytic levels, including document, sentence, aspect, word, character, and sub-word levels. Sentiments posted on social media are an extensively valuable resource for governments, businesses, and other organizations. For instance, companies can monitor the efficacy of their products and services by analyzing feedback received from social media platforms. In addition, they can collect valuable data and business expertise to enhance the development of future products and services. Furthermore, organizations can figure out the level of satisfaction or dissatisfaction that customers have toward their products and services in order to enhance their reputation. Moreover, they can determine potential consumers from the whole audience and conduct market divisions to enhance business decisions.[7,8] Due to the vast amount of online data that includes opinions from individuals, it is not practical to manually analyse this data since this is time-consuming, costly, and subjective,[9] therefore, researchers are focusing on SA.

The Arabic language plays a significant role in social network communication, with an increasing number of users utilizing this language daily. There are more than 400 million people who use it every day, and it is one of the official languages of the UN. While SA has garnered researchers' interest, analyzing sentiments in Arabic content has limitations and is developing slowly compared to languages such as English.[10] The Arabic research limitation is because of various factors: Firstly, the language form can be classified into classical Arabic (CA), modern standard Arabic (MSA), and dialect Arabic (DA). Classical Arabic (CA), the earliest form of Arabic, follows exact morphological and grammatical principles. It is also primarily used for reading literary works, prayers, and the Holy Qur'an. Modern standard Arabic (MSA) is the primary language in the Arab world. It is used in official contexts, including documents, areas for study, and writing books. Dialectical Arabic (DA) is the dialect-based language that individuals use to communicate on a daily basis. It is used in social media posts and conversations from daily life. Geographical and social class differences influence the Arabic dialects. Secondly, the Arabic orthography: Arabic letters change their appearance depending on where in a word they are located (beginning, middle, or end). Nouns are not capitalized, and the writing is from right to left. The majority of Arabs speak their own dialect of Arabic rather than Modern Standard Arabic (MSA). The presence or absence of diacritics can change the meaning of words. On social media, the majority of words are devoid of diacritics, hence enhancing the difficulty of analysis.[11] Thirdly, language morphology: in Arabic, a word can consist of a stem or root alongside one or more affixes. The root can produce various words with various meanings, hence increasing the complexity during language analysis. Therefore, Arabic NLP applications must handle the Arabic language's complexity.

Considering latest developments in deep learning (DL) techniques across several domains, researchers and data scientists have extensively employed them for handling natural language processing tasks, including sentiment analysis. DL models such as recurrent neural networks (RNNs), long short-term memory (LSTM), and bidirectional LSTM (BiLSTM) have the ability to capture meaningful representations of textual input. The word embedding (WE) is an essential element in various DL techniques that has become prevalent due to its ability to augment the efficacy of neural networks and optimize the performance of DL models.[12] The majority of prior Arabic research has predominantly utilized DL models that prioritize context-independent word embedding, such as Word2Vec, GloVe, and fastText. These models assign a consistent representation to each word, irrespective of its contextual usage. When contrasting the Arabic dataset to the English dataset, common techniques face challenges in outperforming in activities related to classification or forecasting. As a result, the Arabic bidirectional encoder representations from transformers (AraBERT) model will be applied for Arabic sentiment analysis. The AraBERT is a pre-trained language model based on Google's BERT architecture for Arabic language processing tasks.[13] Transfer learning (TL) improves classifier performance by enabling knowledge to be transferred from one domain or dataset to others that have not been seen before, allowing the classifier to become more generalizable. It has lately gained popularity among researchers and data scientists due to the drawbacks of traditional ML techniques. The traditional ML algorithms require starting from scratch during training, which is costly computationally and requires a large amount of data to get acceptable performance. Additionally, a separate training approach is employed, wherein every model receives independent training without employing prior knowledge for a specific job. TL

is still successful even when the domains, tasks, and distributions utilized for training and testing are varied.[14] Although DL methods have increased the accuracy of Arabic SA, these approaches can still be improved.[15]

This study chose to develop a classification model for both positive and negative sentiments because neutral sentiment did not indicate whether the service was good or bad; only positive and negative attitudes were chosen. It might also depend on a polite and understanding consumer who treats things normally. Additionally, neutral sentiment merely presents things as normal, whereas telecom companies are more focused on "what was good or bad" specifically. As a result, only the positive and negative sentiment classes were chosen for this study. Therefore, the most significant contributions of this study are as follows: first, to extract features from Arabic text using the AraBERT model. Second, to develop a novel deep neural network that combines the RNN and BiLSTM models, referred to as RNN-BiLSTM, for Arabic sentiment analysis. Third, to optimize the performance of the hybrid model as well as the RNN and BiLSTM models using the hyper-parameter fine-tuning method. Fourth, evaluate the developed approach with an Arabic dataset.

This article has the following sections: Section 2 discusses the literature review. Section 3 outlines the proposed methodology. Section 4 presents the experimental study and results analysis. Section 5 presents the results as well as a discussion of the significant results. Lastly, Section 6 presents a concise conclusion derived from the conducted research and the obtained findings.

2 | LITERATURE REVIEW

2.1 | Models for Arabic sentiment analysis

In recent times, there has been a notable growth in the utilization of social media platforms by individuals for the purpose of expressing their perspectives or providing feedback, either positive or negative, regarding a particular service. Sentiment analysis has grown significantly lately due to the advancements in technology and the abundance of social media data. This can be attributed to the importance of utilizing opinions derived from the analysis of the user's social media material in the process of decision-making. The study of Reference 16 investigated public sentiment toward the recent Monkeypox epidemic using a CNN-LSTM hybrid model on a publicly available dataset of tweets relevant to the epidemic. The hybrid model achieved a significant accuracy of approximately 91% by employing various preprocessing, validation, and encoding procedures. Furthermore, the hybrid model was validated by comparing it to traditional machine learning techniques. The authors in Reference 17 examined the application of deep learning techniques, specifically the LSTM model, in the context of sentiment analysis of customer reviews. The Amazon review dataset was implemented to gain insights into product quality and performance. In addition, a comparative analysis was performed between the deep LSTM model and other traditional machine learning techniques, such as SVM, naive bayes, decision tree, and logistic regression. The LSTM model achieved an accuracy of 93.66%. In Reference 18, the authors studied the classification of reviews among various ethnic groups using real-time data from a social media network. The reviews were categorized based on sentiment polarity, including positive, negative, and neutral. Also, the study used a CNN technique to carry out experiments and compare findings with other machine learning techniques. The results indicated that the CNN model achieved levels of accuracy up to 94.47%, 95.4%, and 94% when evaluating the Phone, Laptop, and TV review datasets, respectively.

ML methods have become prevalent in the field of SA. However, the effectiveness of these methods for handling raw data is limited, and the representation of features has a big impact on how well a ML model performs. Due to its efficacy, the DL technique has garnered significant attention across several domains. Consequently, researchers and data scientists have progressively employed this technique to tackle the challenge of SA. Although current DL algorithms have increased the accuracy of SA in Arabic, these approaches still require improvement.[15]

The researchers in Reference 19 did a sentiment analysis of user reviews for movies by employing three machine learning models: SVM, multinomial Naive Bayes, and Bernoulli classifiers. These models were applied to different datasets. The results of the study showed that the SVM achieved a 90% accuracy rate, outperforming the performance of the other two models.

The authors in Reference 20 employed DL techniques to examine the emotional content of Arabic tweets, and they proposed categorization of statements into four emotions. They did a comparison between the CNN algorithm, ML algorithms, multi-layer perceptron (MLP), SVM, and Naive Bayes (NB) algorithms in the task of emotion classification of

Arabic tweets. The findings of the study indicate that the CNN algorithm demonstrated superior performance compared to the ML algorithms, achieving an accuracy rate of 99.82%.

In Reference 21, many architectures of DL methodologies were devised, including RNNs with LSTM, RNNs with Bi-LSTM, and CNNs. The study employed restaurant reviews obtained from Yelp in order to conduct sentiment categorization using binary and multi-class approaches. The findings indicate that the Bi-LSTM model demonstrated superior performance with regard to accuracy compared to other models, earning a score of 95.76% for binary classification and 64.03% for multi-classification.

In a further investigation conducted by Reference 22, two DL techniques, namely LSTM and Bi-LSTM, were employed to categorize sentiments in texts written in the Saudi Arabian dialect. This study employed a combination of two DL approaches alongside the widely known SVM algorithm. The dataset utilized in this research consisted of 32,063 tweets. The results of their study indicated that the Bi-LSTM model outperformed other models with an accuracy rate of 94%.

The study done by Reference 23 aimed to evaluate the effectiveness of BiLSTM in improving Arabic sentiment analysis. The researchers applied the forward-backward approach to collect contextual information from sequences of Arabic features. The final outcomes from six widely recognized sentiment analysis datasets demonstrate that the proposed model significantly outperforms both contemporary deep learning models and traditional machine learning methodologies used as benchmarks.

Word embedding (WE) is a useful technique for obtaining numerical representations from words. The authors in Reference 24 proposed a model that uses a DL approach to tackle Arabic Sentiment Analysis (ASA). The model was trained using LSTM as a DL network, with word embedding employed as the initial hidden layer for the purpose of feature extraction. The findings indicated that the DL approach had an accuracy rate of approximately 82%. Also, the authors in Reference 25 studied SA in the Arabic language, using Word2Vec and BLSTM. Word representation models are used to transform the words used in reviews into their corresponding vectors. The BLSTM model accepts a sequence of words within sentences as its input. The Word2Vec representation model, which is based on meaning and context, was used to compute the polarity. They presented a DL architecture based on BLSTM. At a maximum accuracy of 94.88, The BLSTM model architecture demonstrates superior performance compared to both the CNN and LSTM models.

Training classical word embeddings from scratch requires a large text corpus and hence takes a considerable amount of time. In the field of natural language processing (NLP), there is a wide range of pre-trained word embedding vectors that are easily accessible to the public. The researchers in Reference 15 developed a combined method using CNN and LSTM models. In addition, they used a pre-trained word-embedding model named AraVec to analyse Arabic tweets. The model outperformed the current DL models by achieving state-of-the-art performance. Specifically, the Arabic Sentiment Tweets Dataset (ASTD) obtained an accuracy of 65.05%. The researchers in Reference 26 introduced a hybrid CNN-LSTM model that uses word2vec embeddings. The researchers employed many Arabic sentiment analysis datasets, namely Main-AHS, Sub-AHS, Ar-Twitter, and ASTD. The Sub-AHS dataset has achieved a remarkable accuracy of 96.8%.

In Reference 27, the authors examined different methods documented in existing literature to address the challenge of sentiment analysis for regional dialects. The authors proposed a methodology utilizing AraBERT word embedding that was designed for sentiment analysis of the Moroccan dialect. In addition, they did a 2-way classification and comparative research of ML algorithms like SVM, DT, LR, RF, NB, and DL algorithms like LSTM, BiLSTM, and LSTM-CNN from the state of the art. As a result, it was shown that BiLSTM achieved better results for both 2-way classification, with an accuracy of 83%, as well as in 4-way classification scoring 62% to 92% accuracy in each of the four classes.

The study conducted by Reference 28 aimed to comprehensively investigate the use of context-independent and context-dependent word embeddings in the field of Arabic sentiment analysis. The implementation involved utilizing pre-trained word embeddings as fixed extraction techniques to provide feature inputs for the CNN model. The findings obtained by conducting tests on two different Arabic datasets demonstrate that the AraBERT model shows the highest effectiveness in performing these tasks. AraBERT achieved a remarkable accuracy rate of 91.4% and 95.49% on the relevant datasets. The study done by the authors in Reference 29 focused on the application of hybrid and DL models for sentiment analysis in Arabic. A novel ensemble-stacking model is suggested, which combines three pre-trained models: the deep layers of CNN, a hybrid model combining CNN and LSTM, and a hybrid model combining CNN and GRU, together with a meta-learner SVM. The suggested model demonstrated better performance compared to existing models, with an accuracy of 92.12% for Main-AHS, 95.81% for Sub-AHS, and 81.4% for ASTD.

The authors of Reference 30 assessed how effectively traditional and contextualized word embeddings performed sentiment analysis activities, employing four widely employed techniques. Classical embedding methodologies encompass GloVe, Word2vec, and FastText, while contextualized models employ ARBERT. Sentiment categorization tasks have employed deep learning architectures such as BiLSTM and CNN. The results of the study indicated that a particular technique consistently outperforms its pre-trained counterpart, with BERT demonstrating the highest level of performance. The BiLSTM model demonstrates superior performance compared to the CNN model, with a 2% higher accuracy across three datasets.

Transfer learning is a methodology that involves applying the knowledge and features acquired from a previously learned model, which allows us to avoid training a new model from scratch. The authors of Reference 31 have presented a deep CNN model for doing SA in Arabic. This model relies on character-level representation as its basis. Furthermore, the use of the model extends to the application of TL in the domains of sentiment analysis and emotion identification, specifically in the Arabic language. The purpose of the program is to utilize Arabic character-level feature representation that has been acquired from a substantial sentiment data set in order to implement Arabic emotion recognition. The results of the implementation demonstrated improved performance in the detection of emotions, as measured by accuracy.

In Reference 32, the authors examined the ability of the AraBERT model to obtain universal contextualized phrase representations with the goal of showing its utility for Arabic text multi-class categorization. AraBERT was used as a TL model and feature extractor. After adjusting its parameters on the OSAC datasets, they employed the AraBERT model to transfer its knowledge for categorizing Arabic text. By combining AraBERT with other classifiers, such as CNN, LSTM, Bi-LSTM, MLP, and SVM, its effectiveness as a feature extractor model was examined as well. Finally, they performed an extensive series of tests evaluating AraBERT and multilingual BERT. The study showed that the AraBERT model achieves up to 99% percent accuracy and an F1-score.

RNN is a type of neural network that retains information about sequences by using hidden states. However, due to the problem of vanishing gradients during backpropagation, learning systems that depend on gradients take an extensive amount of time to train, as noted by Reference 33. This led to the development of LSTM. However, LSTM has some drawbacks; for example, because the sentence is only read in one direction, it does not fully account for post-word information.[34] To address this issue, a BiLSTM language model has been developed. A BiLSTM neural network can combine both past and future sequences to produce output.

Due to the problems with RNN and LSTM and the limitations of BiLSTM, and to take advantage of the strengths of RNN and BiLSTM, this study suggested that the hybrid model RNN-BiLSTM be used as the basic model for transfer learning for Arabic sentiment analysis tasks. Despite the use of transfer learning,[35] mentioned that Arabic sentiment analysis is still an important challenge. Also, the authors in Reference 36 demonstrated the need for additional efforts to implement modernized deep learning methods for Arabic sentiment analysis techniques. And hence other research can be conducted to enhance the performance.

3 | MATERIALS AND METHODS

The following section outlines the tested approach for the application of the suggested model, RNN-BiLSTM, in the context of Arabic sentiment analysis. Initially, an overview of the data pre-processing is provided. Subsequently, the technique of word embedding was employed to transform the textual data into vector representations. After that, the DL algorithms, namely the RNN model, the BiLSTM model, and the RNN-BiLSTM hybrid model, were trained and applied for the purpose of tweet classification. Furthermore, the study used a transfer and deep learning-based hybrid model, specifically the proposed RNN-BiLSTM model, to classify the sentiments expressed in tweets.

3.1 | Data processing

Data pre-processing, referred to as data cleaning, is a crucial phase in data analysis because it is crucial to get rid of unneeded, distracting, or redundant data in order to achieve effective sentiment classification and maximize classification accuracy. This is because the accuracy of the model is greatly influenced by the quality of the input data. The data cleaning process involved the following steps:

1. **Noise removal**: At this stage, unnecessary and insignificant text items were removed to improve classification performance. And it includes the following:

   - Remove URLs, punctuation, numbers, special characters, digits, hashtags, and user mentions (@user).
   - Remove all non-Arabic text, including English words. The goal is to standardize the language.
   - Elongation removal: Take off the repeated character, leaving only one.
   - Replace any emoticon with its meaning.
   - Remove the diacritic "tashkeel".
   - Remove Arabic stop words, these words (e.g., pronouns and prepositions) are not useful in the text categorization.

2. **Tokenization**: Tokenization is an important step in NLP that may have an impact on the SA of texts used in social media, because it decreases the typographical variation of words. The method of tokenizing involves dividing the text into tokens, which are words or other important parts of the text. The tokenization process seeks to identify potential keywords by investigating the words that comprise the sentence.
3. **Normalization**: Normalization refers to the procedure of amalgamating the various shapes of different Arabic letters into one single shape. The process operates in a way that is similar to stemming but at the level of letters. The complexity of Arabic morphology is very high, hence necessitating the process of normalization.
4. **Stemming**: Stemming refers to the procedure of eliminating affixes from words and extracting the basic roots of words. The process involves the removal of any affixes, including prefixes, suffixes, and infixes, that are associated with tweets. The most important objective of stemming tweets is to reduce derived or inflected words to their respective stems, bases, or roots in order to enhance SA. In contrast to the root stemming method, it was shown that light stemming provided perfect results in the context of text categorization issues. Moreover, various research studies have demonstrated the benefits of light stemming.[37,38]

3.2 | Word embedding

Various issues associated with sentiment analysis as well as NLP can be handled with the help of word embedding. Word embedding also known as feature extraction, is a method used in feature learning and language modeling to improve SA. This technique involves learning contextual features and transforming words into real-valued vectors of reduced dimensionality. The aim of word embedding is to imbue words with semantic representation based on their similarities or associations with other words. Word embedding has become regarded as the initial stage in DL approaches. This is due to the fact that DL algorithms are unable to directly handle textual input, necessitating the conversion of text into vector representations through the utilization of various word embedding techniques.

There are a number of widely used word embedding models, including word to vector (Word2Vec), proposed by Reference 39. FastText is developed by Facebook.[40] The global vectors for word representation (GloVe) developed by Reference 41. Another famous model is Bidirectional Encoder Representation from Transformer (BERT), which is a state-of-the-art language model.[42] In this study, the text was transformed into vectors using the AraBERT model as a word-embedding layer.

3.2.1 | AraBERT

Google has introduced a novel pre-trained language model known as the BERT model, which is intended for application in the field of NLP. The BERT model distinguishes itself from prior methods in NLP through its enhanced performance. It achieves this by incorporating a bidirectional approach, considering both left-to-right and right-to-left contexts when analyzing a text string. This methodology effectively enhances the preservation of word representations within sentences, constituting its primary advantage.

AraBERT is a language model that has been pre-trained based on Google's BERT model specifically for the purpose of processing Arabic language challenges. AraBERT has undergone extensive training on a large corpus of Arabic text, resulting in a robust comprehension of the language. Consequently, it exhibits commendable accuracy in a variety of

tasks like sentiment analysis, text classification, and language translation. Due to these capabilities, the study employed AraBERT for vector generation, as shown in Figure 1.

3.3 | Modeling

3.3.1 | Recurrent neural networks model

The sequential relationship between words plays a crucial role in textual sentiment analysis which places a lot of emphasis on it.[43] introduced a language model referred to as recurrent neural networks (RNN) which has gained widespread recognition for its effectiveness in handling sequential textual data. The main feature of RNN is the presence of a hidden state that possesses the capability to retain information for the purpose of processing input sequences that may vary in length. Given the word embedding vectors $(x_1, x_2, \ldots, x_n)$, at each time step t, the neural network receives as inputs the word embedding vector x_t and the previous hidden state $h_t - 1$. Next, it employs an activation function to produce the current hidden output h_t. The RNN model's framework is shown in Figure 2.

The update of recurrent hidden state h_t is implemented as follows:

$$h_t^R = g(U \times x_t + W \times h_{t-1}^R + b), \tag{1}$$

$$O_t^R = g(U^o \times h_t + b^o). \tag{2}$$

In the given case, g represents an activation function, such as a logistic function or a hyperbolic tangent function. Additionally, O_t^R denotes the output or predicted value derived by a recurrent neural network. The recurrent hidden state of the network at each time step t is h_t^R determined by the input vector x_t, the previous hidden state h_{t-1}^R, and the bias b. The weight matrices, denoted as W and U, work as filters that determine the degree of significance to be

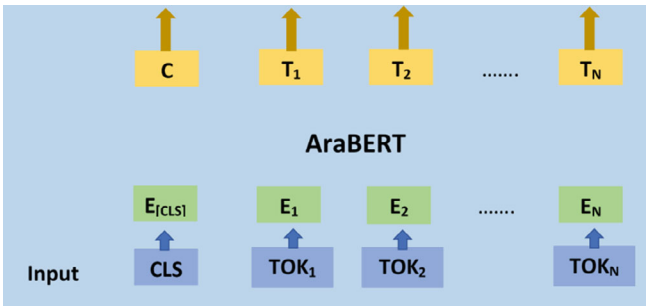

FIGURE 1 The AraBERT architecture.

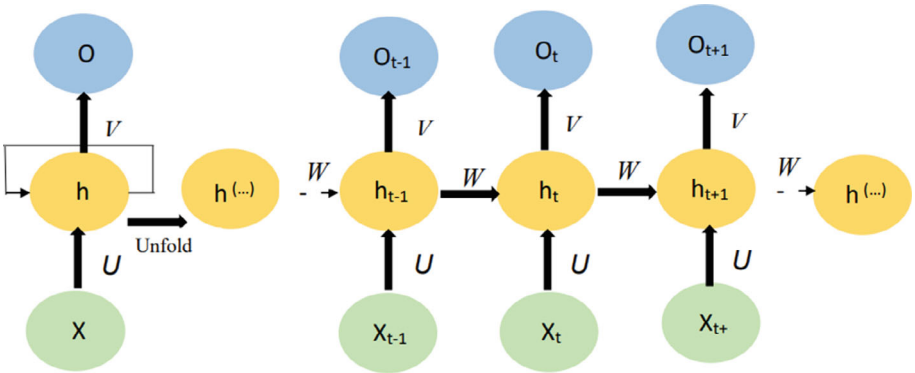

FIGURE 2 The framework of RNN.

given to the input at present and the previous hidden state. They generate an error, and this is then sent via back propagation and used to adjust the weights till the error reaches a minimum threshold and further reduction becomes unattainable.

3.3.2 | Bi-directional long short-term memory model (BiLSTM)

The long-short-term memory (LSTM) network, introduced by Hochreiter and Schmidhuber in Reference 44, is a special variant of the RNN. The model has the ability to acquire and retain information regarding long-term dependencies while effectively addressing the issues of gradient explosion or vanishing. The LSTM unit is composed of a memory cell c_t, an input gate i_t, a forget gate f_t, and an output gate o_t, as shown in Figure 3. These gates are employed for the purpose of storing and regulating the information originating from the cell at a certain time interval. The LSTM model exclusively propagates information in the forward direction. This leads to the computation of an output vector that depends only on the present input at time t and the output of the previous unit.

The long short-term memory (LSTM) model's equations are as follows:

$$i_t = \sigma(W^i \times x_t + U^i \times h_{t-1}^L + b^i), \tag{3}$$

$$f_t = \sigma(W^f \times x_t + U^f \times h_{t-1}^L + b^f), \tag{4}$$

$$g_t = tanh(W^g \times x_t + U^g \times h_{t-1}^L + b^g), \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \tag{6}$$

$$o_t = \sigma(W^o \times x_t + U^o \times h_{t-1}^L + b^o), \tag{7}$$

$$h_t^L = o_t \odot tanh(c_t), \tag{8}$$

where f_t denotes the forget gate, i_t refers to the input gate, g_t is an activation function, c_t denotes the memory cell state, o_t is the output gate, h_t is the regular hidden state, σ indicates a sigmoid function, and $\odot$ denotes element-wise multiplication.

The BiLSTM proposed by Schuster and Paliwal[46] is a technique that enables bidirectional storage of input sequences in neural networks, allowing for information flow in both forward and backward directions. This distinguishes it from the normal LSTM, which only allows for unidirectional input flow. This technique is a development of the RNN methodology.

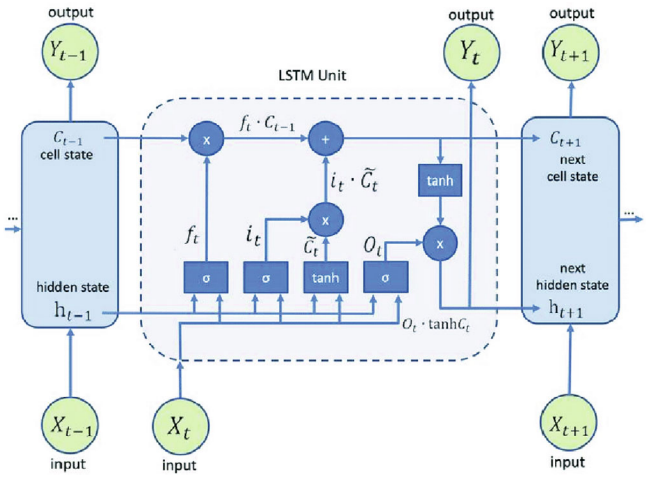

FIGURE 3 The LSTM model architecture.[45]

BiLSTM has achieved state-of-the-art results in many different fields, including natural language processing, handwriting recognition, phoneme classification, and speech recognition. The hidden state *ht* of BiLSTM at time *t* includes both forward h_t^f and backward h_t^b, which are then combined by concatenating their outputs at each time step to generate the final cell output as shown in Figure 4. The computation of the final state of the h_t^{BiLSTM} involves concatenating the two hidden states as follows;

$$h_t^f = tanh(W_{xh}^f \times x_t + W_{hh}^f \times h_{t-1}^f + b_h^f), \tag{9}$$

$$h_t^b = tanh(W_{xh}^b \times x_t + w_{hh}^b \times h_{t+1}^b + b_h^b), \tag{10}$$

$$h_t^{BiLSTM} = h_t^f + h_t^b, \tag{11}$$

where h_t^f and h_t^b are the hidden layer values in the forward and backward directions, respectively, W_{xh} is the weight from the current neuron's input x to the hidden layer h_t at this time, W_{hh} is the weight from the previous state quantity to the current state quantity, h_{t-1} is the hidden layer's output value at the previous moment, and b_h is a bias vector. We used BiLSTM since it provides access to the long-term context in both input and output directions, as well as full learning of the specific problem.

3.3.3 | The proposed hybrid RNN-BiLSTM base model

This section provides comprehensive details about our proposed hybrid model, as depicted in Figure 5. RNN and BiLSTM are used together to create the hybrid model. According to the studies conducted by References 48 and 49, it has been demonstrated that both of these neural network models showed superior performance compared to classification-based approaches in the context of accuracy for SA predictions.

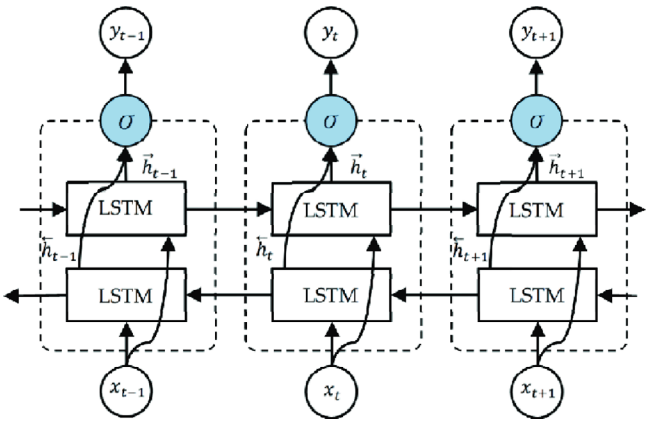

FIGURE 4 The Bi-LSTM model architecture.[47]

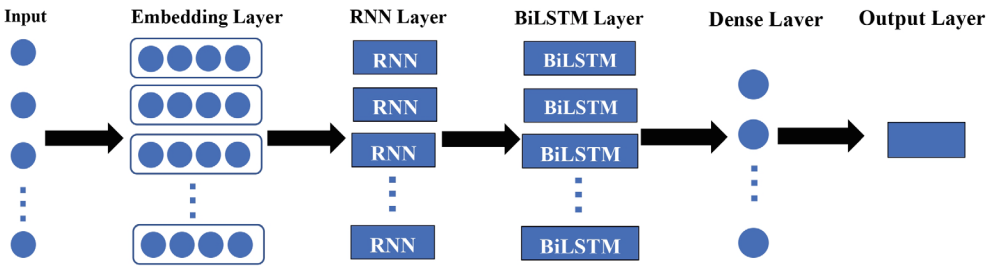

FIGURE 5 The RNN-BiLSTM model architecture.

The primary reason for the utilization of these networks is the fact that the prediction of customer reviews constitutes a challenge that may be approached using classification methods. In the beginning phase, following the pre-processing of input sentences, the pre-processed data was used as input for the AraBERT model. The vector embeddings were generated by the AraBERT layer. The embedding vectors were inputted into the RNN model, resulting in the generation of the output O_t^R, as depicted in Equation (2). In order to obtain the final prediction $\hat{y}_t$, the RNN layer's output is subsequently fed to the BiLSTM layer.

In mathematical terms, the input to the LSTM layer will shift from x_t to O_t^R as depicted below,

$$f_t^H = \sigma(W^f \times O_t^R + U^f \times h_{t-1}^L + b^f), \tag{12}$$

$$i_t^H = \sigma(W^i \times O_t^R + U^i \times h_{t-1}^L + b^i), \tag{13}$$

$$g_t^H = tanh(W^g \times O_t^R + U^g \times h_{t-1}^L + b^g), \tag{14}$$

$$c_t^H = i_t^H \times g_t^H + f_t^H \times c_{t-1}^H. \tag{15}$$

$$o_t^H = \sigma(W^o \times O_t^R + U^o \times h_{t-1}^L + b^o), \tag{16}$$

$$h_t^H = o_t^H \times \tanh(c_t^H). \tag{17}$$

But because this study used BiLSTM, the differences will be in the hidden layer. So for the BiLSTM model, the hidden layers are given by:

$$h_t^F = \tanh(W_{xh}^f \times O_t^R + W_{hh}^f \times h_{t-1}^f + b_h^f), \tag{18}$$

$$h_t^B = \tanh(W_{xh}^b \times O_t^R + w_{hh}^b \times h_{t+1}^b + b_h^b), \tag{19}$$

$$H_t = (w_{hy}^f \times h_t^F + w_{hy}^b \times h_t^B + b_y). \tag{20}$$

And finally, the predicted class $\hat{y}_t$ is given by;

$$\hat{y}_t = \sigma(H_t), \tag{21}$$

where w_{hy}^f and w_{hy}^b are the weights matrix of the output gate and h_t^F and h_t^B are the hidden states of the BiLSTM unit given by Equations (18) and (19).

3.3.4 | Transfer learning

In contrast with traditional approaches in ML and data mining, which make the assumption that training and testing data originate from the same feature spaces and distributions, transfer learning (TL) provides the capability to address situations where domains and distributions show differences. TL is a methodology that involves leveraging a pre-trained model and utilizing its acquired knowledge to address a distinct yet interconnected problem. This process entails transferring the learned insights from prior tasks, thereby enhancing performance and diminishing the duration required for training. In our study, we utilized a source domain dataset to train the basic model and subsequently transferred the knowledge acquired from it to the target domain datasets.

4 | EXPERIMENTAL RESULTS

The experimental settings are described comprehensively in this section, datasets, configuration and evaluation metrics employed in the study.

4.1 | Datasets

In this paper, three sentiment analysis datasets are used in the experiment to show the flexibility of our model across different domains and sizes. The Arabic reviews dataset (ARD) was used as the source domain for our model. On the other hand, the Arabic sentiment tweets dataset (ASTD) and the gold standard corpus (GSC) AraCust Dataset were used as target domains.

4.1.1 | The Arabic reviews dataset

The Arabic reviews dataset (ARD),[50] consists of an extensive collection of Arabic reviews. The ARD dataset contains a wide variety of topics, making it a comprehensive resource for understanding the sentiments and opinions of Arabic-speaking individuals. It includes opinions from 100,000 customers, covering many aspects such as hotels, films, books, and other subjects, including selected airlines. It is categorized into two main types: negative and positive.

4.1.2 | The Arabic sentiment tweets dataset

The Arabic sentiment tweets dataset (ASTD),[51] is a collection of around 10,000 tweets obtained from Twitter, specifically designed for sentiment analysis purposes. The tweets are categorized into four distinct categories: objective, subjective positive, subjective negative, and subjective mixed. Since we are specifically focused on the positive and negative classes, we excluded the objective and neutral classes. Hence, out of the chosen tweets, 1682 were classified as positive, while 797 were classified as negative.

4.1.3 | The gold standard corpus AraCust dataset

The gold standard corpus (GSC) AraCust dataset,[52] consists of 20,000 customer reviews of Saudi telecoms companies. It includes 7590 tweets from STC, 5950 tweets from Zain, and 6450 tweets from Mobily, all of which were written in Arabic. The authors collected this dataset from Twitter and manually labeled each tweet as either positive or negative. However, the dataset is unbalanced, with 6433 positive classes and 13567 negative classes.

4.2 | Experimental setup

The experimental setup's computer has the following specifications: a Windows 11 operating system, an Intel (R) Core i7-1260p processor at 2.80 GHz, 16 GB of RAM, and an NVIDIA GeForce RTX^{TM} 2050 laptop GPU (4 GB GDDR6) graphics card. The research employed Google's TensorFlow v2.9.1 for its implementation. It is an open-source tool designed for constructing machine learning models. It leverages the higher-level API of Keras, which is built upon TensorFlow. This framework also incorporated the open-source DL library for Python, enabling the creation and deployment of machine learning models.

4.3 | Performance evaluation metrics

In order to evaluate the performance of the proposed model, it is important to show the confusion matrix and calculate specific performance metrics. These metrics assist in comparing and evaluating the performance of the proposed method with other methodologies used by researchers in the literature. The confusion matrix is a matrix used to determine the correctness of classification algorithms. Several evaluation metrics can be obtained from the confusion matrix. Accuracy, recall, precision, and F1 score are the four performance metrics used to evaluate the proposed model. The equations representing these evaluation metrics are as follows;

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}, \tag{22}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{23}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{24}$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{25}$$

where TP (true positive) is the number of classifications that are expected to be positive and are correctly predicted. TN (true negative) is the number of classifications that are expected to be negative and are correctly predicted. FP (false positive) is the number of classifications that are expected to be positive, but are incorrectly predicted as negative. FN (false negative) is the number of classifications that are expected to be negative, but are incorrectly predicted as positive.

5 | RESULTS AND DISCUSSION

5.1 | Results

This section presents the results of this study that implemented RNN, BiLSTM, and RNN-BiLSTM models on the source and target domains.

5.1.1 | Baseline models (with the default parameters)

In this section, we present the results of training the hybrid model RNN-BiLSTM using the source domain dataset. The training was conducted using default parameters. To enhance the model's performance, we employed the k-fold cross-validation technique during the training process. This approach involves dividing the dataset into K parts, with each fold serving as a validation set at a certain stage, giving K accuracy as a result. The mean of the model's performance across all folds was then calculated to get an overall result. Figure 6 shows the confusion matrices calculated for the three different models. The model's performance was evaluated based on metrics such as accuracy, precision, recall, and F1-score. The proposed model was also compared with baseline models like RNN and BiLSTM, which were trained on the same dataset. The results are shown in Table 1.

5.1.2 | Transfer learning performance

In this section, we explore the impact of transfer learning by fine-tuning the hyper-parameters of the hybrid RNN-BiLSTM model. This process involved updating the model's weights using the validation dataset from the target domain. We

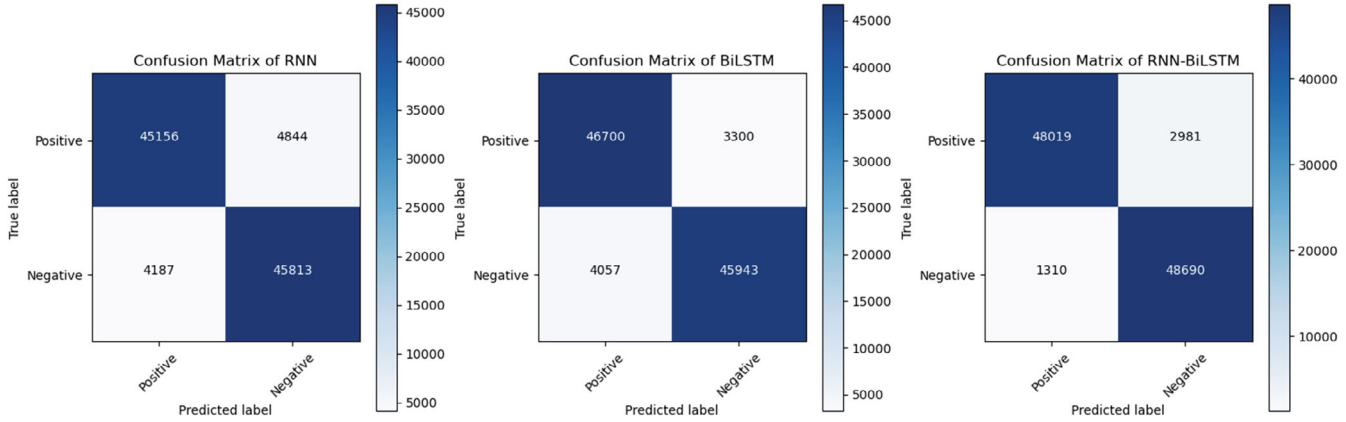

FIGURE 6 Confusion matrix for RNN, BiLSTM, and RNN-BiLSTM models on source domain dataset.

TABLE 1 Results acquired from the baseline models.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RNN	90.97	90.31	91.51	90.91
BiLSTM	92.64	93.40	92.01	92.70
RNN-BiLSTM	95.75	94.15	97.34	95.72

TABLE 2 Fine-tuned RNN-BiLSTM model parameters configuration.

Hyper-parameter	Value
Number of layers	5-6
Activation function	Sigmoid
Dropout rate	0.2
Learning rate	0.0001
Loss function	Categorical cross-entropy
Number of epochs	20
Batch size	64
Optimizer	Adam

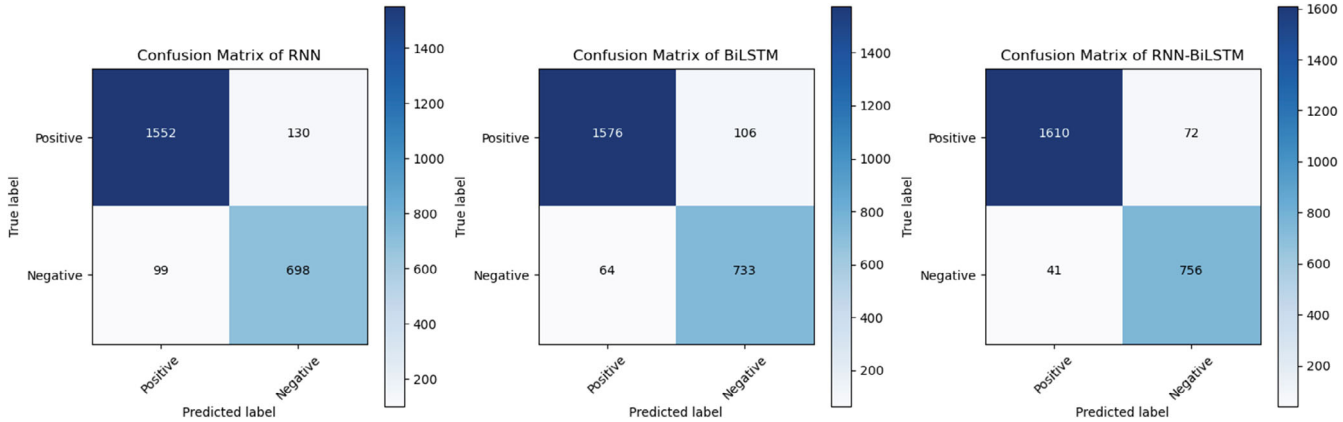

FIGURE 7 Confusion matrix for RNN, BiLSTM, and RNN-BiLSTM models on ASTD dataset.

employed a random search approach to identify the most effective hyper-parameter setup for optimization purposes. We used early stopping as a strategy to determine the optimal number of epochs for training the model, which helps conserve computational resources, prevent over-fitting, and demonstrate strong generalization capabilities without excessive training. The first layers of the base model were frozen to facilitate the training of the target model for classification. This approach was adopted to preserve valuable learned representations. Subsequently, a dense layer was added to the frozen layers, and we proceeded to train the model using 64 batches and 20 epochs. Table 2 presents the best hyper-parameter configuration that achieved the highest level of accuracy. Figure 7 and 8 shows the confusion matrices calculated for the three different models. Figure 9 illustrates a consistent improvement in the model's accuracy over several epochs. We compared the proposed model with the same baseline models as in the previous section. These models were designed similarly to the proposed RNN-BiLSTM hybrid model. The results are shown in Table 3 for the testing dataset from the target domain.

5.2 | Discussion

This section discussed the results and the strengths and weaknesses of our proposed hybrid RNN-BiLSTM model for sentiment analysis.

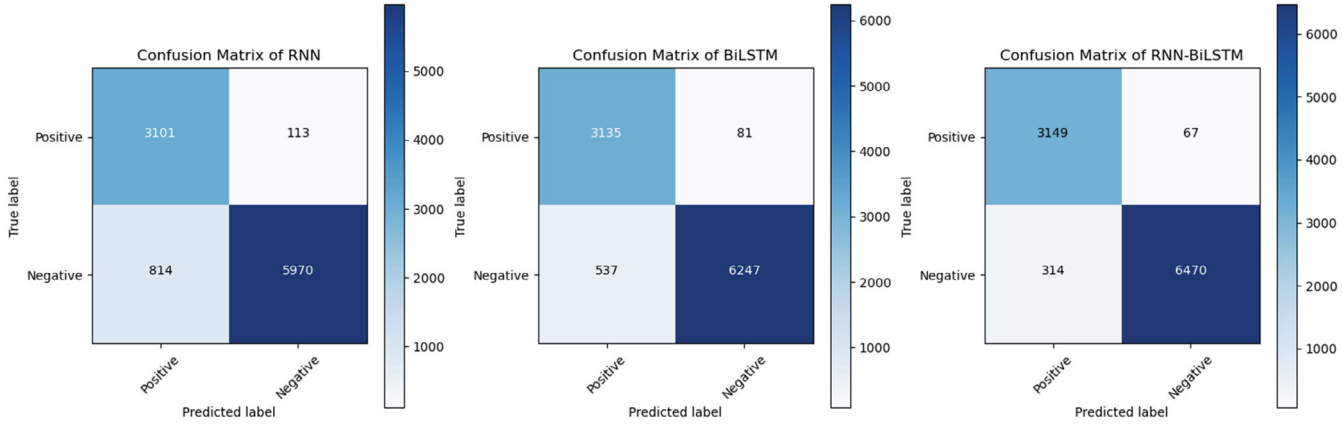

FIGURE 8 Confusion matrix for RNN, BiLSTM, and RNN-BiLSTM models on AraCust dataset.

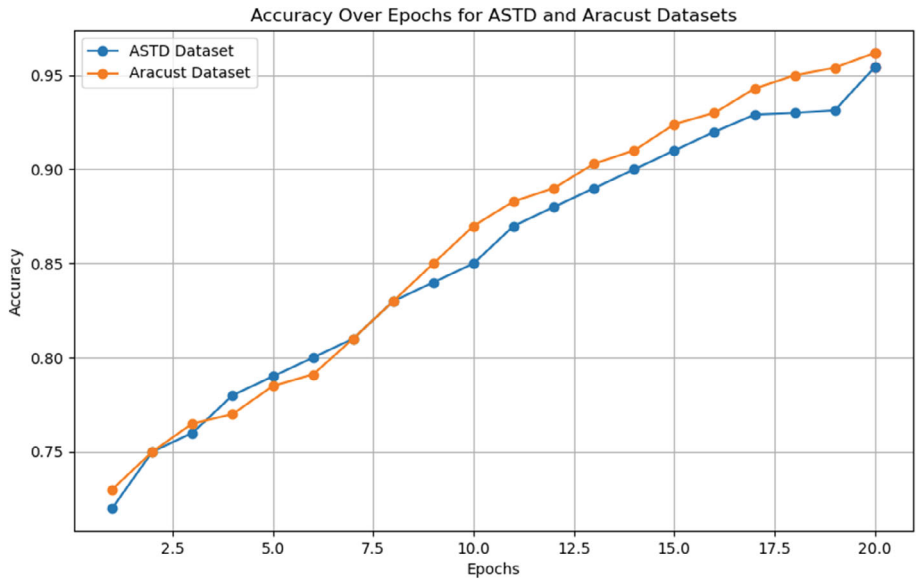

FIGURE 9 Accuracy over epochs for ASTD and Aracust datasets.

TABLE 3 Results acquired after fine-tuned.

Dataset	Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
ASTD	RNN	90.76	92.27	94.00	85.84
	BiLSTM	93.14	93.70	96.10	94.88
	RNN-BiLSTM	95.44	95.72	97.52	96.61
Aracust	RNN	90.73	96.48	79.21	87.00
	BiLSTM	93.82	97.48	85.38	91.03
	RNN-BiLSTM	96.19	97.92	90.93	94.30

We compared the performance of the proposed RNN-BiLSTM model and its components, which are the RNN and BiLSTM models. We evaluated the models on a source domain and a target domain, where the target domain has two different datasets. We used accuracy, precision, recall, and F1-score as the evaluation metrics. The source domain "the Arabic Reviews Dataset (ARD)" is a collection of Arabic reviews covering many aspects. The target domain has two different datasets: ASTD and Aracust for sentiment analysis.

Table 1 shows the results of the three models on the source domain. The proposed RNN-BiLSTM model outperforms the other two models with the default parameters. The RNN-BiLSTM model achieves an accuracy of 95.75%, a precision of 94.15%, a recall of 97.34%, and an F1 score of 95.72%. In comparison, the individual RNN model achieved an accuracy of 90.97%, precision of 90.31%, recall of 91.51%, and an F1 score of 90.91%. The BiLSTM model, on the other hand, achieved an accuracy of 92.64%, precision of 93.40%, recall of 92.01%, and an F1 score of 92.70%. This indicates that the RNN-BiLSTM model can learn both short-term and long-term dependencies in the text and can capture the sentiment of the reviews more accurately than the other two models.

The RNN-BiLSTM model was fine-tuned on the validation dataset from the Aracust dataset. The fine-tuning process aimed to adapt the model to the target domain, which has different vocabulary and syntax than the source domain. Table 3 shows the results of the fine-tuned RNN-BiLSTM model on the two target datasets: ASTD and Aracust. The fine-tuned RNN-BiLSTM model shows consistent and high performance on both target datasets. On the ASTD dataset, the RNN-BiLSTM model achieves an accuracy of 95.44%, a precision of 95.72%, a recall of 97.52%, and an F1 score of 96.61%. On the Aracust dataset, the RNN-BiLSTM model achieves an accuracy of 96.39%, a precision of 98.32%, a recall of 91.15%, and an F1 score of 94.60%. This suggests that the RNN-BiLSTM model can generalize well to different domains and tasks and can identify the relevant aspects more effectively than the other two models.

Employing its capacity to retain sequential information through hidden states, the RNN model demonstrated a certain level of effectiveness in text classification tasks. On the ASTD and Aracust datasets, the model recorded accuracy of 90.76% and 90.73%, precisions of 92.27% and 96.48%, recalls of 94.00% and 79.21%, and F1 scores of 85.84% and 87.00%, respectively. However, the model's significantly lower recall rate of 79.21% on the Aracust dataset suggests potential difficulties in identifying positive instances within this specific dataset. This could be due to the vanishing gradient problem in the RNN model, which makes it difficult to learn long-term dependencies in the text. On the other hand, the BiLSTM model, designed to capture both forward and backward dependencies in the data, demonstrated enhanced performance. It achieved accuracy of 93.14% and 93.82%, precisions of 93.70% and 97.48%, recalls of 96.10% and 85.38%, and F1 scores of 94.88% and 91.03% on the ASTD and Aracust datasets, respectively. This indicated its improved capability to understand the context and sequence of words in the text. While the BiLSTM model's recall rate surpasses that of the RNN model, it is lower than the RNN-BiLSTM model on both target datasets.

Our results demonstrate that the RNN-BiLSTM model is a powerful and versatile model for text classification and that it can adapt well to different domains and tasks. Moreover, the integration of RNN and BiLSTM models allows for the capture of both short-term and long-term dependencies in the data. Furthermore, the use of the k-fold cross-validation strategy helps to minimizing over-fitting and enhancing the performance of the model. However, there are some limitations and challenges that need to be addressed. For instance, the RNN-BiLSTM model is more computationally expensive and time-consuming than the other two models, and it may require more data and resources to train and optimize. Moreover, the RNN-BiLSTM model may not be able to handle more complex and diverse text data, such as multilingual, or multi-modal text. The model was not trained on sarcasm or indirect expressions; therefore, its ability to accurately classify sarcastic or indirect sentiments may be limited. Therefore, future work could explore ways to enhance the efficiency and scalability of the RNN-BiLSTM model and to incorporate more features and techniques to deal with more challenging text data.

5.3 | Comparison with previous studies

The aim of this section is to compare the performance of the proposed model with some existing models. The results presented in Table 4 indicate that the proposed model showed better results than the techniques examined in previous work, except for the CNN-LSTM method in Reference 26. Compared to the ensemble approach of CNN and LSTM proposed by Reference 15, which obtained an accuracy of 65.05% on the ASTD dataset, our proposed RNN-BiLSTM model showed better performance, achieving an accuracy of 95.75% on the same dataset. The model proposed by Reference 24 that uses LSTM for Arabic Sentiment Analysis (ASA) achieved an accuracy rate of approximately 82%. The accuracy of

TABLE 4 Comparison between Proposed RNN-BiLSTM with previous studies.

Reference	Class	Approach	The best result
15	3	CNN-LSTM	65.05%
26	2	CNN-LSTM	96.8%
24	2	LSTM	82%
29	2	Stacking SVM based on integrated: CNN, CNN-LSTM, CNN-GRU	95.81%
25	2	CNN, LSTM, and BiLSTM	BiLSTM: 94.88%
30	2,3, and 4	BiLSTM, CNN	BiLSTM: 93.47%
Our proposed model	2	RNN-BiLSTM	96.18%

the stacked SVM based on integrated CNN, CNN-LSTM, and CNN-GRU was 79.18% in Reference 29. In Reference 25, the authors used CNN, LSTM, and BiLSTM models; the BiLSTM achieved 94.88% accuracy. In Reference 30, the accuracy of the Bi-LSTM model was recorded at 93.47%.

5.4 | Error analysis

When examining cases of inaccurate predictions, it is clear that the most common errors happened when a text included both positive and negative terms. In addition, there are misspellings and grammatical mistakes. Overall, the texts of the positive class contain important phrases or words that may belong in the negative class or vice versa, potentially causing misclassification and giving worse results.

6 | CONCLUSIONS

This study introduces a hybrid RNN-BiLSTM model for Arabic sentiment analysis, leveraging the power of transfer learning. The model employs AraBERT for the extraction of word embedding vectors from textual reviews, thereby capturing the contextual relationships between words. Next, this contextual information is fed to the RNN to remember sequence information through hidden states. The output of the RNN is then passed to the BiLSTM layer to capture sequential features. The model was trained on a source domain dataset, achieving an accuracy of 95.75%. In addition, the hyper-parameters of the proposed model were fine-tuned to explore the impact of transfer learning and enhance the overall classification performance. Following this fine-tuning, the model was tested on a target domain dataset, where it achieved accuracy rates of 95.44% and 96.19% for the ASTD and Aracust datasets, respectively. Despite the fact that the RNN-BiLSTM model outperformed most of the previous methods, there remains potential for further improvement. Future work will aim to utilize a variety of feature extraction techniques to enhance the performance of deep learning. Extending this model for multilingual sentiment analysis by incorporating multilingual datasets while addressing the unique challenges of each new language. It is also recommended to consider sentiment at the character level rather than the sentence level to improve results.[53] In addition, explore datasets that include sarcastic and indirect expressions and train the model on these datasets to improve its performance in handling such expressions. Furthermore, the model could be enhanced to recognize a broader range of sentiments. In conclusion, while the RNN-BiLSTM model has demonstrated promising results, continuous efforts to improve its efficiency and versatility will ensure its applicability to a wider range of tasks and datasets in the field of Arabic sentiment analysis.

AUTHOR CONTRIBUTIONS

Conceptualization: D.B., L.N., and A.N. *Formal analysis*: D.B. *Methodology*: D.B., L.N., and A.N. *Software*: D.B. *Supervision*: L.N. and A.N. *Validation*: D.B. *Visualization*: D.B. *Writing—original draft preparation*: D.B. *Writing—review and editing*: L.N. and A.N. All authors have read and agreed to the published version of the manuscript.

ACKNOWLEDGMENTS

The first author would like to thank the Pan African University, Institute for Basic Sciences, Technology, and Innovation (PAUSTI), Nairobi, Kenya, and the African Union. Additionally, I am incredibly grateful to my family and friends for their continuous support, encouragement, and understanding during this journey. Your confidence in me got me through hard times.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interests.

PEER REVIEW

The peer review history for this article is available at https://publons.com/publon/10.1002/eng2.12874.

DATA AVAILABILITY STATEMENT

The datasets used in this study can be accessed through the dataset citations.

ORCID

Duha Mohamed Adam Bakhit https://orcid.org/0009-0001-2366-549X

REFERENCES

1. Borgman CL, Scharnhorst A, Golshan MS. Digital data archives as knowledge infrastructures: Mediating data sharing and reuse. *J Assoc Inf Sci Technol*. 2019;70(8):888-904.
2. Yu M, Li Z, Yu Z, He J, Zhou J. Communication related health crisis on social media: a case of COVID-19 outbreak. *Curr Issue Tour*. 2021;24(19):2699-2705.
3. Sarwar R, Zia A, Nawaz R, Fayoumi A, Aljohani NR, Hassan SU. Webometrics: evolution of social media presence of universities. *Scientometrics*. 2021;126:951-967.
4. Jahangir M, Afzal H, Ahmed M, Khurshid K, Nawaz R. An expert system for diabetes prediction using auto tuned multi-layer perceptron. *2017 Intelligent Systems Conference (IntelliSys)*. IEEE; 2017:722-728.
5. Edara DC, Vanukuri LP, Sistla V, Kolli VKK. Sentiment analysis and text categorization of cancer medical records with LSTM. *J Ambient Intell Humaniz Comput*. 2023;14(5):5309-5325.
6. Yang S, Xing L, Li Y, Chang Z. Implicit sentiment analysis based on graph attention neural network. *Eng Rep*. 2022;4(1):e12452.
7. Mehta P, Pandya S, Kotecha K. Harvesting social media sentiment analysis to enhance stock market prediction using deep learning. *PeerJ Comput Sci*. 2021;7:e476.
8. Lo SL, Chiong R, Cornforth D. Ranking of high-value social audiences on Twitter. *Decis Support Syst*. 2016;85:34-48.
9. Alharbi FR, Khan MB. Identifying comparative opinions in Arabic text in social media using machine learning techniques. *SN Appl Sci*. 2019;1(3):213.
10. Mohammed A, Kora R. Deep learning approaches for Arabic sentiment analysis. *Soc Netw Anal Min*. 2019;9:1-12.
11. Boudad N, Faizi R, Thami ROH, Chiheb R. Sentiment analysis in Arabic: A review of the literature. *Ain Shams Eng J*. 2018;9(4):2479-2490.
12. Prottasha NJ, Sami AA, Kowsher M, et al. Transfer learning for sentiment analysis using BERT based supervised fine-tuning. *Sensors*. 2022;22(11):4157.
13. Antoun W, Baly F, Hajj H. Arabert: Transformer-based model for Arabic language understanding. arXiv preprint, arXiv:2003.00104 2020.
14. Lu J, Behbood V, Hao P, Zuo H, Xue S, Zhang G. Transfer learning using computational intelligence: A survey. *Knowl-Based Syst*. 2015;80:14-23.
15. Heikal M, Torki M, El-Makky N. Sentiment analysis of Arabic tweets using deep learning. *Proc Comput Sci*. 2018;142:114-122.
16. Mohbey KK, Meena G, Kumar S, Lokesh K. A CNN-LSTM-based hybrid deep learning approach for sentiment analysis on Monkeypox tweets. *N Gener Comput*. 2023;1-19.
17. Mohbey KK. Sentiment analysis for product rating using a deep learning approach. *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*. IEEE; 2021:121-126.
18. Meena G, Mohbey KK, Indian A. Categorizing sentiment polarities in social networks data using convolutional neural network. *SN Comput Sci*. 2022;3(2):116.
19. Mohbey KK. A comparative analysis of sentiment classification approaches using user's opinion. *Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)*. 2018:26-27.
20. Baali M, Ghneim N. Emotion analysis of Arabic tweets using deep learning approach. *J Big Data*. 2019;6:1-12.
21. AlSurayyi WI, Alghamdi NS, Abraham A. Deep learning with word embedding modeling for a sentiment analysis of online reviews. *Int J Comput Informat Syst Ind Manage Appl*. 2019;11:227-241.
22. Alahmary RM, Al-Dossari HZ, Emam AZ. Sentiment analysis of Saudi dialect using deep learning techniques. *2019 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE; 2019:1-6.
23. Elfaik H, Nfaoui EH. Deep bidirectional LSTM network learning-based sentiment analysis for Arabic text. *J Intell Syst*. 2020;30(1):395-412.

24. Al-Bayati AQ, Al-Araji AS, Ameen SH. Arabic sentiment analysis (ASA) using deep learning approach. *J Eng*. 2020;26(6):85-93.
25. Elsamadony OM, Keshk AE, Abdelatey A. Arabic language sentiment analysis using bidirectional long short term memory. *IJCI Int J Comput Informat*. 2023;10(1):65-77.
26. Al Omari M, Al-Hajj M, Sabra A, Hammami N. Hybrid CNNs-LSTM deep analyzer for Arabic opinion mining. *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE; 2019:364-368.
27. Matrane Y, Benabbou F, Sael N. Sentiment analysis through word embedding using AraBERT: Moroccan dialect use case. *2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA)*. IEEE; 2021:80-87.
28. Zouidine M, Khalil M. A comparative study of pre-trained word embeddings for Arabic sentiment analysis. *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE; 2022:1243-1248.
29. Saleh H, Mostafa S, Gabralla LA, O. Aseeri A, El-Sappagh S. Enhanced Arabic sentiment analysis using a novel stacking ensemble of hybrid and deep learning models. *Appl Sci*. 2022;12(18):8967.
30. Sabbeh SF, Fasihuddin HA. A comparative analysis of word embedding and deep learning for Arabic sentiment classification. *Electronics*. 2023;12(6):1425.
31. Omara E, Mosa M, Ismail N. Emotion analysis in Arabic language applying transfer learning. *2019 15th International Computer Engineering Conference (ICENCO)*. IEEE; 2019:204-209.
32. Fz E-A, El Alaoui SO, Nahnahi NE. Contextual semantic embeddings based on fine-tuned AraBERT model for Arabic text multi-class categorization. *J King Saud Univ Comput Informat Sci*. 2022;34(10):8422-8428.
33. Hochreiter S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int J Uncertain Fuzzi Knowled Based Syst*. 1998;6(2):107-116.
34. Cliche M. BB_twtr at SemEval-2017 task 4: Twitter sentiment analysis with CNNs and LSTMs. arXiv preprint, arXiv:1704.06125 2017.
35. Alqarni A, Rahman A. Arabic Tweets-Based Sentiment Analysis to Investigate the Impact of COVID-19 in KSA: A Deep Learning Approach. *Big Data Cognitive Comput*. 2023;7(1):16.
36. Nassif AB, Elnagar A, Shahin I, Henno S. Deep learning for Arabic subjective sentiment analysis: Challenges and research opportunities. *Appl Soft Comput*. 2021;98:106836.
37. Aldayel HK, Azmi AM. Arabic tweets sentiment analysis–a hybrid scheme. *J Inf Sci*. 2016;42(6):782-797.
38. 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(8):43-49.
39. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. *Adv Neural Inf Proces Syst*. 2013;26.
40. Joulin A, Grave E, Bojanowski P, Mikolov T. Bag of tricks for efficient text classification. arXiv preprint, arXiv:1607.01759 2016.
41. Pennington J, Socher R, Manning CD. Glove: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics; 2014:1532-1543.
42. Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint, arXiv:1810.04805 2018.
43. Mikolov T, Karafiát M, Burget L, Cernockỳ J, Khudanpur S. Recurrent neural network based language model. *Interspeech*. Vol 2. Makuhari; 2010:1045-1048.
44. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput*. 1997;9(8):1735-1780.
45. Lazaris A, Prasanna VK. An LSTM framework for software-defined measurement. *IEEE Trans Netw Serv Manag*. 2020;18(1):855-869.
46. Schuster M, Paliwal KK. Bidirectional recurrent neural networks. *IEEE Trans Signal Proces*. 1997;45(11):2673-2681.
47. Li YH, Harfiya LN, Purwandari K, Lin YD. Real-time cuffless continuous blood pressure estimation using deep learning model. *Sensors*. 2020;20(19):5606.
48. Bemila T, Kadam I, Sidana A, Zemse S. An approach to sentimental analysis of drug reviews using RNN-BiLSTM model. *Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST)*. 2020.
49. Sangeetha J, Kumaran U. A hybrid optimization algorithm using BiLSTM structure for sentiment analysis. *Measure Sens*. 2023;25:100619.
50. Arabic 100k Reviews. https://www.kaggle.com/datasets/abedkhooli/arabic-100k-reviews
51. Nabil M, Aly M, Atiya A. ASTD: Arabic sentiment tweets dataset. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics; 2015:2515-2519.
52. Almuqren L, Cristea A. AraCust: A Saudi Telecom Tweets corpus for sentiment analysis. *PeerJ Comput Sci*. 2021;7:e510.
53. Elhassan N, Varone G, Ahmed R, et al. Arabic sentiment analysis based on word embeddings and deep learning. *Compute*. 2023;12(6):126.

How to cite this article: Bakhit DMA, Nderu L, Ngunyi A. A hybrid neural network model based on transfer learning for Arabic sentiment analysis of customer satisfaction. *Engineering Reports*. 2024;e12874. doi: 10.1002/eng2.12874