Hindawi
Mobile Information Systems
Volume 2022, Article ID 1068554, 11 pages
https://doi.org/10.1155/2022/1068554

Research Article

Sentiment Prediction of Textual Data Using Hybrid ConvBidirectional-LSTM Model

Dashrath Mahto[1] **Subhash Chandra Yadav**[1] **and Gotam Singh Lalotra**[2]

[1] *Computer Science and Technology, Central University of Jharkhand, Ranchi, India*
[2] *Department of Computer Science, Govt. Degree College Basohli, Basohli, India*

Correspondence should be addressed to Dashrath Mahto; dashrathmahto12@gmail.com

Received 4 April 2022; Accepted 25 May 2022; Published 20 June 2022

Academic Editor: M. Praveen Kumar Reddy

With the emergence of social media platforms, most people have changed their way of interacting. Perhaps, sharing day-to-day lifestyle updates is a trend substantially influenced by microblogging sites, specifically Twitter, Facebook, Instagram, and many more. Moreover, text and messages are the most preferred way for such interactions. Twitter is one of the most commonly used microblogging tools that enable people to express their thoughts, opinions, emotions, happiness, sadness, excitement, ideas, mental stress, and so on. Hence, the sentiment prediction furnished by such textual data becomes a complex and challenging task. In this research, the authors proposed a hybridization of the convolutional neural network and bi-directional long short-term memory model (named ConvBidirectional-LSTM), which aims to better the categorization of sentiments of text data. Then, this proposed hybrid ConvBidirectional-LSTM model is compared with the existing state-of-the-art models, GloVe-based CNN-LSTM and Hierarchical Bi-LSTM (HeBiLSTM) models. Furthermore, the performance of the proposed hybrid ConvBidirectional-LSTM model is evaluated on the US airline dataset using various performance parameters like accuracy, precision, recall, and f_1 score. The proposed model outperformed the existing state-of-the-art models with an accuracy rate of 93.25% in sentiment prediction.

1. Introduction

The tremendous growth of Internet utilization, particularly by microblogging websites, has resulted in the generation of a significant volume of textual information conveying peoples' choices, thoughts, and emotions. This textual information is significantly helpful and might be used by businesses, governments, and others to make choices; evaluation tools that can extract necessary knowledge from it and categorize it are depend on its polarity. This problem is explored in the area of sentiment classification, which is a branch of computational linguistics [1]. The microblogging movement emerged due to the fast evolution of information and communication technologies, which has increased online media users around the globe. Web 2.0's major characteristics include collaborative information exchange and users on a digital site, resulting in massive volumes of unstructured material on various themes. As a result, microblogging sites became necessary to share connections and business interactions. The increasing use of social media platforms like Twitter, LinkedIn, Facebook, and consumer reviews media has sparked interest in the Microblogging era. With the rapid growth of social mass communication and textual content, sentiment classification employing customer reviews attracts enormous interest from various organizations (e.g., commercial and academic) [2]. In the cyberspace, still, a variety of research is prevailing using data mining approaches for social networking text data; significant examples include connectivity, material, and customer data [3]. One information that has become a substantial focus of recent studies is detecting individuals' opinions in blog articles about a particular topic, referred to as sentiment prediction. In this research work, three sentiment labels, positive, negative, and neutral, are used. Twitter, a prominent social network site, facilitates users with a tool that any individual can send and receive short text messages. A distinctive feature of Twitter allows it to appeal to businesses, including its transparency, its word restriction on uploaded posts, and the widespread utilization of hashtags. Even though most social media platforms need two members to be connected before they can even see anyone's posts, Twitter enables people to see each other's posts even if they do not know each other; it is simple to gather information [4].

1.1. Sentiment Analysis. Every tweet contains either a positive, negative, or neutral sentiment. The sentiment of the user can be determined using the sentiment score thatis calculated based on the positive and negative words in a tweet, as shown in the following equation [5]:

$$\text{sentiment score} = \frac{(P - N)}{(P + N + 2)}. \tag{1}$$

Here, P and N define the total count of positive and negative words in a tweet, respectively. The sentiment score is represented using a discrete 2-valued variable S that represents the sentiment class:

$$S \in \{-1, 1\}. \tag{2}$$

All the sentiment score values and the differences between them are captured by the variable S. In some cases, the polarity value fails to identify the degree of sentimentality from the textual data because, in some instances, the negative and positive sentiment scores cancel each other which results in a zero sentiment score, i.e., (sentiment score = 0). Though the textual data from the tweet are positive or negative and not neutral, the zero sentiment score results in false data. Hence, the following constraints are followed to identify positive and negative tweets.

$$\begin{aligned} S &= 1\,(\text{positive tweet})\text{if sentiment score} \geq 0.1 \\ &= -1\,(\text{negative tweet})\text{if sentiment score} < 0.1. \end{aligned} \tag{3}$$

When the tweets are provided as input to the system, the polarity of that tweet is calculated to identify whether the given tweet is positive, negative, or neutral.

1.1.1. Applications of Sentiment Classification. Some of the prominent applications of sentiment classification are given in the section as follows:

(i) Mainly used for classifying sentences, paragraphs, and documents into positive, negative, or neutral labels.

(ii) Used in commercial applications such as multimedia systems for writing movie reviews [6], news articles, restaurant reviews, mobile customer reviews [7], real-time insights, etc. [8].

(iii) Used to extract meaning from the sentence, classification of intent, and linguistics-based emotion analysis.

(iv) Used by product manufacturing companies for obtaining accurate product reviews based on customer ratings.

(v) Used in some of the fields where sentiment prediction is promptly needed like Emotion Detection (E.D.), Building Resources (B.R.), Transfer Learning (T.L.), etc.

(vi) Used in creating artificial datasets by utilizing semi-supervised machine learning algorithms.

1.2. Motivation. Sentiment prediction from textual data has gained vast significance. The prominent attributes procured from sentiment analysis can be used in decision-making, psychological processes, opinion collection for political promotion, product marketing, and so on. Due to enormous textual data generation, social media platforms have become prominent data sources for sentiment prediction-related works. Primarily, microblogging sites such as Twitter are widely used to collect people's opinions and views in "tweets" that have a maximum length of 140 characters. The anonymity of Twitter makes it easy for people to express their original sentiments on the microblogging site. Several approaches are being used to analyze text sentiments from Twitter data. Previous methods used for sentiment prediction are based on sentiment lexicons and need to restrict themselves to external resources or manual preprocessing for complex feature analysis. The anomalies related to the existing approaches motivated the authors to propose an enhanced deep neural network model that can extract different sentimental features from textual data and can accurately predict people's sentiments.

1.3. Research Objectives. The primary objective of this proposed work is to develop an effective sentiment prediction model based on textual data to analyze different sentiments. The proposed approach comprises a hybrid ConvBidirectional-LSTM model for extracting both word sequence and word semantic features for sentiment prediction.

The main objective of the proposed research work is mentioned in the following section:

(i) This research article aims to improve the accuracy of the previous work carried out by different researchers [9–13]. The authors proposed a hybrid ConvBidirectional-LSTM approach for analyzing different sentiments with the US airline dataset and predicting the polarity of the text sentiments. The ConvBidirectional-LSTM is the hybridization of the CNN and Bi-LSTM to learn complex contextual features and semantic information from the airline Twitter data.

(ii) The experimental analysis lies on the real dataset collected from Twitter used for predicting positive, negative, and neutral sentiments.

(iii) To portray the Twitter posts in the format of integer data or arrays, a pretrained GloVe (https://github.com/stanfordnlp/GloVe accessed on 26 February

2022) word embedding approach will be employed. This approach is pretrained unlabeled word matrices that could preserve word meaning and learn with a massive set of words. This research will use a GloVe embedding vector approach to examine the quality of the proposed framework.

(iv) The results of the ConvBidirectional-LSTM model will be compared with the preexisting state-of-the-art models, experimental GloVe-based CNN-LSTM, and HeBiLSTM models [14, 15] to verify the efficacy of the proposed framework.

This research paper contains related research background in Section 2, the proposed hybrid ConvBidirectional-LSTM for sentiment prediction in Section 3, research experiment in Section 4, and results and discussion in Section 5, and finally the research work is concluded in Section 6.

2. Background

This section reviews the related work and existing models used for sentiment analysis and also introduces word embedding, convolutional neural networks, and bidirectional-LSTM approaches.

2.1. Related Research. Sentiment classification, widely called opinion mining, draws the attention of the customers using text mining and NLP approaches [16]; research has found that people potential, worker surveillance, and real-time insights [17] are all advantages of sentiment classification [18]. Network operators can use sentiment classification to find out what type of services they are missing and what areas of their existing customers are happy [19]. Sentiment classification operates by distinguishing positive and negative thoughts inside text evaluations, which can be quite difficult to recognize inside delicate wordplays [20]. Table 1 lists several related sentiment classification research studies.

Basiri et al. [9] suggested an attention-based bidirectional CNN-RNN deep model (ABCDM) with eight different datasets such as App, Movies, Kindle, US-airline, Electronics, CDs, Sentiment140, and T4SA. The ABCD Model uses two BiLSTM and GRU phases to retrieve future and past contents in both directions. In addition, the attention mechanism is used on the outcomes of the bidirectional stages of ABCDM to focus on particular terms. Therefore, the max-pooling phase is utilized to minimize feature dimensionality by extracting contextual information. Further, the ABCDM model was compared with state-of-the-art models and with attention-based CNN and BiLSTM (AC-BiLSTM), SS-BED, HAN, ARC, CRNN, and IWV. This research article considered positive and negative sentiments for their experiments. Jain et al.,[11] proposed a CNN-LSTM model for sentiment analysis of the US airline and US airline quality dataset. This research only considered positive and negative sentiments for their experiments and achieved 91.3% accuracy with CNN-LSTM on the US airline dataset. This paper compares the proposed CNN-LSTM model with existing machine algorithms such as Support Vector Classifier (SVC), Decision Tree (D.T.), Logistic Regression (L.R.), and Naive Bayes (N.B.), CNN, and LSTM. Umer et al. [12] provides a deep learning (DL) network that combines a CNN and LSTM for sentiment prediction on Twitter data. And then, this CNN-LSTM model was compared with existing ML algorithms [33–26], specifically with the SVC, Random Forest (R.F.), L.R., SGD, a Voting Classifier of SGD, and R.F. Moreover, the sentiment prediction performance is examined with two influential text extracting features, TF-IDF and word2vec. The authors evaluated the overall performance of the proposed method using three different datasets like the US airline, hate speech, and women's e-commerce clothing reviews. Sezgen et al., [27] used the Latent Semantic Analysis (LSA) [28] text mining technique for sentiment analysis [29]. For that very purpose, they collected 2,536 negative and 2,584 positive airline reviews data from TripAdvisor.com. The study is focused on examining the fundamental factors that determine passenger dissatisfaction and satisfaction and the variations among airlines' business strategies. The intrinsic shortcoming of bag-of-words data analysis is that the LSA classifier does not evaluate sentence-level certain text content deriving from a grammatical structure. Xu et al. [30] suggested a BiLSTM-based sentiments data analysis for posts and used it to address the problem of post-sentiment classification. The opinion knowledge participation intensity is incorporated into the TF-IDF technique of phrase strength calculation. A group-led approach of basis vectors based on modified phrase strength calculation is suggested in response

Table 1: Recent related work for sentiment analysis on the US airline dataset.

Dataset	Authors	ML algorithm
US airlines	Umer et al. [12]	LR, VC SGD, SVC RF, LSTM CNN-LSTM
US airlines, Airline quality	Jain et al. [11]	NB, LR DT, SVC LSTM, CNN CNN-LSTM
US airlines	Monika et al. [21]	RNN, LSTM
US airlines	Rustam et al. [22]	SVC, LR RF, DT, CC SGD, GNB GBC, ETC AdaBoost, VC GBM, LSTM
US airlines	Kumar and Zymbler [23]	SVC, ANN, CNN
US airlines	Hakh et al. [24]	AdaBoost DT, SVC NB, RF K-NN
US airlines	Acosta et al. [25]	SVC, LR

CC: calibrated classifier [26–28], VC: voting classifier, SVC: support vector classifier, DT: decision tree, ANNs: artificial neural networks [29], RNN: recurrent neural network, LSTM: long short-term memory, GNB: Gaussian naive Bayes, K-NN: K-nearest neighbor [30–32], ETC: extra trees classifier, NB: naïve Bayes, GBM: gradient boosting machine, RF: random forest, LR: logistic regression, and SGD: stochastic gradient descent.

to the shortcomings of a phrase developed mathematically in recent studies. Furthermore, the BiLSTM captures all semantic features and can generate decent writing of the remarks. This research paper compared its proposed BiLSTM architecture with existing Naive Bayesian, CNN [31], RNN, and LSTM [32]. Lastly, the opinion characteristic of a message is determined using a co-evolutionary network with SoftMax maps. The correctness of the suggested word representing approach in this research is demonstrated through experimentation with various word representations.

2.2. Word Embedding. Word embedding methods [42–44] are decentralized word representation techniques that use artificial neural network (ANN) structures to express texts into a low-dimensional density matrix of actual values. The ANN model is a ground-breaking research project that trains word embeddings from word occurrences. Word embedding techniques are a powerful way to extract lexical and pragmatic contexts from massive text data. Word2vec [43] is a well-known tool for learning word representations that try to anticipate targeted utterances on contexts using a single-layer ANN architecture. Pennington et al. [44] is a word embedding technique that creates word representations by probing the actual co-occurrence patterns of the text corpora utilizing global statistics.

2.3. Convolutional Neural Networks. CNN is among the most common deep learning techniques [45] that can accurately recognize data in a variety of situations. The CNN model can successfully handle several issues in image processing and intelligent computational linguistics, including sentiment classification, query responses, text summaries, and so on. It is distinguished by a specific architectural design that supports learning. A CNN model is a multilayered network in which one layer's result becomes the input for the next layer. The CNN architecture contains three layers: convolution, max-pooling, and fully connected. Vector V can be subdivided into $\{V_{1:m}, V_{2:m+1}, \ldots, V_{n-m+1:n}\}$ if the convolutional kernel size equals n and it moves vertically and horizontally into vector V in stages of 1. Here, $V_{j:k}$ denotes the linearly connected vector from V_j to V_k. By performing a convolution upon every element $V_{j:k}$, the vectors $D = \{d_1, d_2, \ldots, d_{m-n+1}\}$ get produced, where d_j stands for the mapping of local features, and the equation is given as follows:

$$d_j = f\left(w.V_{j:j+n-1} + b\right). \qquad (4)$$

Here, f is the activation function, w is the weight parameter of the convolution kernel, and b is the bias. The text characteristics map-based vector $D = \{d_1, d_2, \ldots, d_{m-n+1}\}$ collected using convolution is then pooled, and also the study employs the max-pooling approach. The formula is written as follows:

$$\widehat{d}_n = \max\{D\}. \qquad (5)$$

The previous findings were achieved using convolution and max-pooling operations on a single convolution kernel. The following are the findings for c convolution kernels:

$$\widehat{D_n} = \left\{D_{n_1}, D_{n_2}, \ldots, D_{n_c}\right\}. \qquad (6)$$

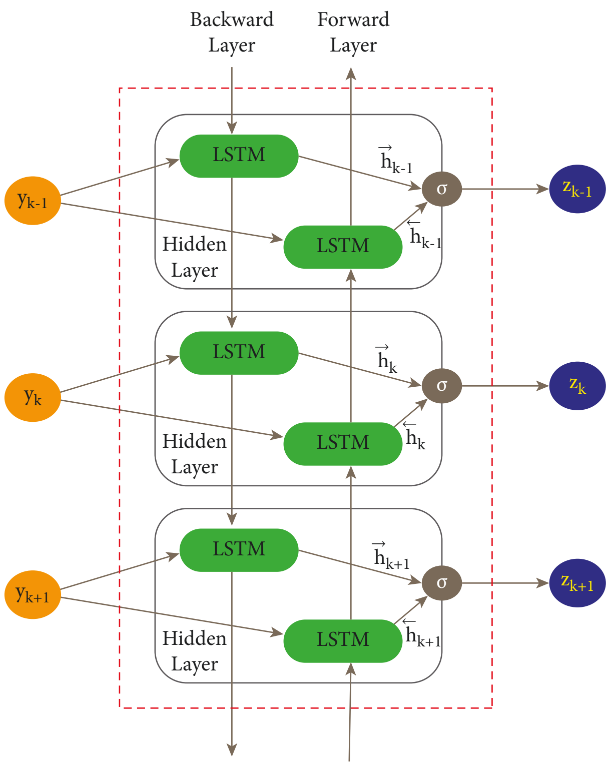

Figure 1: Fundamental structure of the bidirectional LSTM.

After successive convolution and max-pooling layers, the fully connected layer is sent as a complete connection. And finally, this completely linked layer is mapped with a tagged sampling field, and characteristics are merged.

2.4. Bidirectional LSTM. Figure 1 depicts the fundamental structure of the bidirectional LSTM [46].

The Bi-LSTM replicates the primary, recurrent layer in the architecture so that two layers are formed side-by-side. The Bi-LSTM provides the input sequence to the first layer as it is and provides the second layer with a reversed copy of the input sequence. In each iteration, it records the last words stored in its memory unit and evaluates the probability of the next word [47]. For each word stored in the library, the Bi-LSTM allocates a likelihood based on past terms and identifies the word that holds the higher probability, and in its memory, that word is stored. The exemplary memory of the Bi-LSTM makes it suitable for language generation as it remembers the background of the conversation at any moment. The limitation of the traditional Recurrent Neural Networks (RNN) approach to storing long-length word sequences is overcome by the Bi-LSTM. A four-layer neural network is a Bi-LSTM model, and each LSTM's memory unit consists of three gates: the input, output, and forget gates, as shown in Figure 2.

These gates allow the model to either retain or forget words at any moment by regulating the flow of data through that gate. This enables the Bi-LSTM model to track only

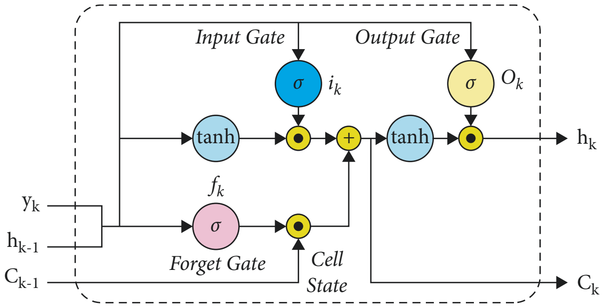

Figure 2: Fundamental structure of the LSTM.

relevant data. This reduces the issue of data disappearing gradient that helps the system remember data stored for a longer time. The cell C_k runs through the network, and the LSTM gates, such as the input gate and output gate, control the flow of data through the Bi-LSTM via the sigmoid function [48]. When the value of the sigmoid activation function [49] is "1," the data are ultimately passed through the gates, whereas if the value of the sigmoid function is "0," the information is not allowed by the entrance. The amount of data that has to be passed through is decided by the forget gate, as defined in [49, 50]

$$f_k = \sigma\left(W_f.\left[h_{k-1}, y_k\right] + b_f\right). \tag{7}$$

Here, σ is the sigmoid activation function, W_f is the input word sequence, h_{k-1} is the previous state of the forget gate, and b_f represents the bottleneck features. Once the data are controlled by the forget gate, the input gate controls the new data that will be retained in cell state C_k. The formula for the input gate is

$$i_k = \sigma\left(W_i.\left[h_{k-1}, y_k\right] + b_i\right). \tag{8}$$

The current state of the cell and the memory unit status of the cell are obtained from equations (9) and (10), respectively.

$$\widehat{C}_k = \tanh\left(W_c.\left[h_{k-1}, y_k\right] + b_c\right), \tag{9}$$

$$C_k = f_k.C_{k-1} + i_k.\widehat{C}_k. \tag{10}$$

Lastly, the output gate in the Bi-LSTM will be used for controlling the output of the sigmoid function as shown in equation (11) and the hidden layer output is defined according to equation (10).

$$O_k = \sigma\left(W_o.\left[h_{k-1}, y_k\right] + b_o\right), \tag{11}$$

$$h_k = O_k.\tanh\left(C_k\right). \tag{12}$$

The terms b and W represent the bias value and the weight coefficient matrix, respectively; tanh is defined as the hyperbolic tangent activation function.

The Bi-LSTM consists of an additional layer of reverse LSTM (Backward LSTM) that reverses the flow of information. The hidden layer synthesizes the forward and backward information. Doing so ensures that every cell in the LSTM can obtain the context information. The reverse or the backward layer of the Bi-LSTM is evaluated similarly to that of the forward LSTM. However, the direction of the information flow is reversed to acquire the following information at a particular time. The forward and backward information flow in the Bi-LSTM network is illustrated in the following equations [51]:

$$\begin{aligned} h_f &= f\left(W_{f_1}.y_k + W_{f_2}.h_{k-1}\right), \\ h_b &= f\left(W_{b_1}.y_k + W_{b_2}.h_{k+1}\right), \end{aligned} \tag{13}$$

where h_f and h_b are the outputs of the forward and backward LSTM, respectively. And the final output of the hidden layer is given as

$$z_i = g\left(W_{o_1} * h_f + W_{o_2}.h_b\right). \tag{14}$$

3. Proposed Hybrid ConvBidirectional-LSTM for Sentiment Prediction

This research uses the ConvBidirectional-LSTM model to construct a revolutionary method for improving sentiment categorization on Twitter messages. This proposed approach integrates the CNN and BiLSTM neural networks. We used this hybridization to see how well the CNN might respond to Bidirectional LSTM, renowned for its efficacy in sentiment prediction. The most significant advantage of this proposed approach is that it enables a large amount of data to be extracted successfully. In the first stage of the model, the word embedding matrix is processed through the **GloVe** embedding technique, and this word vector is then provided as an input to the CNN. In the next stage, the convolutional layer extracts the relevant features. Then, the dimensionality of the feature space for each input text is reduced using the max-pool approach. At last, the features vector is created in the fully linked layers for feature integration. Then, it would be passed to Bidirectional LSTM to get the contextual information about the textual data, which significantly enhances the sentiment classification accuracy of textual data. The suggested ConvBidirectional-LSTM model is described in six subsections as depicted in Figure 3.

3.1. Word Vectorization.

The proposed model takes the text data input in this layer and then splits it into words or tokens. Every word is turned into a numerical integer matrix. The vector of numerical values was produced using the GloVe pretrained word embedding algorithm. The GloVe technique is utilized separately to assess the performance of the model. If every text of m words is expressed as $V = \{w_1, w_2, \ldots, w_m\}$, then every word is transformed into an n-dimensional vector representation, and the text input is specified as

$$V = \{w_1, w_2, \ldots, w_m\} \in R^{m*n}. \tag{15}$$

Due to the variant size of the input vector, the size of the text content **s** used in the proposed model must be equal. If the text content is smaller than **s**, then the text content size gets increased by applying a zero-padding technique, and if

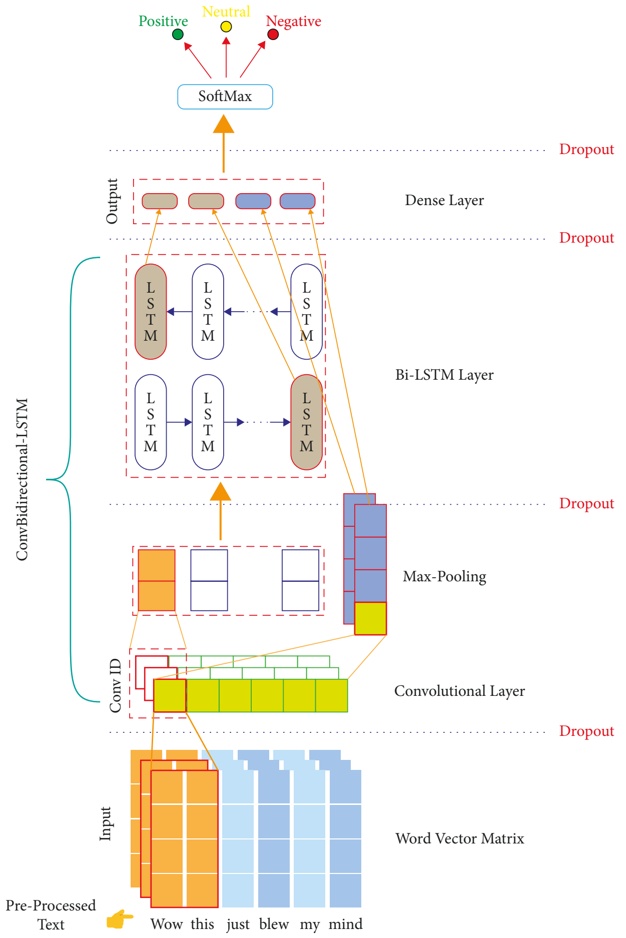

FIGURE 3: Structure of the proposed ConvBidirectional-LSTM model.

the text content is larger than the specified size **s**, it will then be shortened; as a result, every text content has the same vector size. The representation of each text content of **s** dimension is as given in [52]

$$V = \{w_1, w_2, \ldots, w_m\} \in R^{s*n}. \tag{16}$$

3.2. Convolutional Layer. The convolutional layer is an essential stage of the CNN architecture, transcendent for retrieving the relevant features from the Twitter text [53]. For convolution operation, one-dimensional convolutional layer receives the text vector array $V \in R^{s*n}$ from the GloVe embedding layer. To create the feature space of n-gram in a 1-dimensional convolutional layer, an array matrix is being computed from convolutional text using M filtration and the size **p** of a convolutional kernel. The filters F_m, where $1 \leq m \leq M$, produce a feature space depicted in the following equation:

$$d_j^m = f\left\{w^m \otimes Y_{j:j+w-1} + b^m\right\}, \tag{17}$$

where f is the nonlinear activation function, and in this research the swish [54] activation function is used; F_m is the weight vector of filter expressed as $w \in R^{s*n}$; b^m is the bias of filter F_m; n is the word matrix dimension size; and the convolutional operation is represented as $\otimes$; $Y_{j:j+p-1}$ that defines filter F_m retrieves feature $Y_{j:j+p-1}$ from Y_j, and the result of the feature space of filter F_m is d_j^m; here, j^{th} is the component of d^m. The features associated with the text of size s are as in

$$d = \left\{d_1, d_2, \ldots, d_j, d_s\right\}. \tag{18}$$

3.3. Max-Pooling Layer. The convolutional process generates characteristic maps, and thereafter the max-pooling layer collects the significant characteristics $\hat{d} = \max\{d\}$ to compute the local appropriate statistics. One-dimensional max-pooler decreases the dimensions of its source by converting every kernel size into a single result of the highest limit. Thus, the CNN architecture can efficiently minimize the number of features to avoid over-fitting problems while simultaneously reducing run-time and parametric complexities.

3.4. Bi-LSTM Layer. Unlike the LSTM classifier, Bidirectional-LSTM includes two hidden states that allow data to be transferred in both paths forward-to-backward (h_f) and backward-to-forward (h_b). It also helps Bidirectional-LSTM to understand the situation entirely. All source data that comprise both past and future values would be kept using both paths, whereas the typical RNN framework declined to include future trends. The basic implementation of Bidirectional-LSTM is to link two different ways of an LSTM model to a single output. A forward LSTM phase obtains past data, while the reverse LSTM phase obtains future data. This architecture supports the system in retaining past and future data. In Bidirectional-LSTM, the sequential result of the first phase seems to be the input for the next step, while the sequence outcomes of the next step are concatenated with the final unit result of the backward and forward actions. The resulting outcome h after stacked Bidirectional-LSTM layers is shown as in

$$h = \left\{h_f, h_b\right\}. \tag{19}$$

3.5. Dense Layer. The proposed framework incorporates a dense network layer to link each source input with every outcome by utilizing its weights. Softmax as an activation function is used inside the last layer to obtain the final result. Softmax will take the mean value of random outcomes into 0, 1, and 2 forms. Equation (20) represents the predicted outcome of the softmax function. The output of negative, neutral, and positive sentiments is labeled with 0, 1, and 2, respectively, using categorical cross-entropy.

$$Z = \text{soft}\max\{wh + b\}. \tag{20}$$

3.6. Regularisation. Deep learning visualizes overfitting as a most challenging problem. The data seem to be trained efficiently by the model, but failure is observed in the case of unseen data generalization. The regularisation technique can easily avoid over-fitting problems. A regularisation is an

Table 2: Sentiment proportion for each airline in the US airline dataset.

Airlines	Sentiment		
	Negative	Neutral	Positive
American	1960	463	336
Delta	955	723	544
Southwest	1186	664	570
US airways	2263	381	269
United	2633	697	492
Virgin America	181	171	152

approach to improving the generalization ability by making minor changes to the training algorithm. The method adds more prediction models throughout the training to decrease its complexities and avoid over-fitting. Dropout and L2 are the most often used regularisation techniques. L2 regularisation, often referred to as weight decaying or ridge regression, penalizes its loss function by adding the square intensity of the parameter as a punishment. A dropout is an approach for avoiding over-fitting and generalizing the system that involves periodically dropping a component off (both hidden and shown) throughout learning. The architecture of the ConvBidirectional-LSTM model reflects four dropout levels between the word vector matrix and the convolutional, max-pooling, and Bi-LSTM layers, and before and after the dense phase, with a dropout rate of 30%.

4. Research Experiment

Experiments during the research were conducted to test the ConvBidirectional-LSTM text sentiment categorization classifier with the US airline sentiment. This section comprises detailed descriptions of the dataset, data preprocessing, experimental setup, hyper-parameters setting, and performance evaluation metrics.

4.1. Dataset. The *US airline sentiment dataset* has a total number of 14,640 rows and 15 columns; in the underlying research work, only two columns are being considered for the analysis. The first column contains a sentiment label, and another contains a text review of the passengers. In this dataset, labels are categorized into negative, positive, and neutral sentiments with 9178 rows as negative, 3099 as neutral, and 2363 as positive sentiments. The proposed model is trained and tested with 9516 and 5124 data rows. This US airline sentiment dataset is available at Kaggle (https://www.kaggle.com/welkin10/), and also available at Crowd Flower data library (https://www.data.world/crowdflower/). The proportion of sentiments for each airline contained in the US airline dataset [55] is illustrated in Table 2.

4.2. Experimental Setup. Deep learning models can be developed using various methods, tools, and packages. In the proposed research work, Keras [56] is used; it is known to be one of the best tools with TensorFlow as the backend. The research experiments are executed in Google collaboratory and Jupiter notebook on Microsoft Windows 11 P C. with AMD Ryzen3-3250U/Radeon Graphics processor (2.60 GHz) and 8 GB RAM as hardware and software support. To closely correlate the experimental outcomes of the proposed model with other existing state-of-the-art models, the accuracy value is used as a primary performance indicator. Additional performance indicators, including recall, precision, and F_1-measure, are also tested to measure the proposed classifier.

Table 3: Hyperparameter setting.

Parameters	Values
Embedding dimensions	GloVe = 300
Kernel size	5
Filter	128
Activation	Swish
Pool size	3
Dropout	0.3
Bi-LSTM output size	512
Batch size	128
Epoch	15
Optimizer	Adam
Learning rate	0.001
Loss function	Categorical_crossentropy

4.3. Data Preprocessing. To select the essential features from the text data for sentiment prediction, data preprocessing is considered the initial phase of the experiment. Since most of the text data contain a mix of misspelled words, parts-of-speech (POS) tagging, slang, exclamation marks, acronyms, punctuation marks, etcthehe key objective of the preprocessing stage is to remove excessive textual information, conflict, distortions, and inconsistency. In most of the textual data, it has been observed that the text data may be present as straightforward, distorted, inconsistent, and doubtful. Therefore, textual data preprocessing is required to evaluate the similarities and keep them in a format suitable for further investigation. In the underlined research work, the authors have applied the data preprocessing phase to remove stop words and signatures of the airlines, preserving some essential words of the text data such as “nor,” “not,” “no,” etc., which are prone to reflect negative sentiments. Then, stemming and lemmatization functions are applied to that textual data. Further, regular expression (re) was used to correct word repetition and data cleaning to remove username, punctuation, HTML, emoji, and URLs. Furthermore, all the text data have been changed to lowercase to make the text dataset more universal. Finally, the data preprocessing stage tokenized all the text data to particular utterances, and 12041 unique tokens have been retrieved from the US airline sentiment text dataset.

4.4. Hyperparameter Setting. Table 3 shows the values of the hyperparameters that are used in the proposed ConvBidirectional-LSTM model.

4.5. Evaluation Metrics. The performance evaluation indicators listed below are the most important criteria for measuring the performance of the proposed approach.

Table 4: Comparison of the experimental outcomes obtained from the US airline dataset.

Method	Sentiment class	Precision	Recall	F_1	Accuracy (%)
GloVe-CNN-LSTM	Negative	0.99	1.00	1.00	92.79
	Neutral	0.84	0.84	0.84	
	Positive	0.79	0.78	0.78	
	Weighted average	0.93	0.93	0.93	
GloVe-HeBiLSTM	Negative	0.99	1.00	1.00	92.47
	Neutral	0.81	0.84	0.83	
	Positive	0.78	0.73	0.75	
	Weighted average	0.93	0.93	0.93	
Proposed ConvBiLSTM	Negative	**1.00**	**1.00**	**1.00**	**93.25**
	Neutral	**0.86**	**0.85**	**0.85**	
	Positive	**0.80**	**0.80**	**0.80**	
	Weighted average	0.93	0.93	0.93	

$$\begin{aligned} \text{Pre} &= \frac{\text{TRP}}{\text{TRP} + \text{FLP}}, \\ \text{Rec} &= \frac{\text{TRP}}{\text{TRP} + \text{FLN}}, \\ \text{Acc} &= \frac{\text{TRP} + \text{TRN}}{\text{TRP} + \text{TRN} + \text{FLP} + \text{FLN}}, \\ f_1 - \text{score} &= \frac{2 * \text{Pre} * \text{Rec}}{\text{Pre} + \text{Rec}}, \end{aligned} \tag{21}$$

where Pre, Rec, Acc, TRN, TRP, FLN, and FLP are precision, recall, accuracy, true negative, true positive, false negative, and false positive.

5. Results and Discussion

The proposed ConvBidirectional-LSTM framework has experimented with the optimal hyper-parameters setting and compared with the existing state-of-the-art models, GloVe-based CNN-LSTM and HeBiLSTM models. Table 4 shows the overall accuracy, precision, recall, and F_1 score of different deep learning models. The performance accuracy of the proposed ConvBidirectional-LSTM framework was 93.25%, whereas GloVe-CNN-LSTM and GloVe-HeBiLSTM models' performance accuracy was 92.79% and 92.47%, respectively. The observed performance accuracy reflects that the proposed framework outperforms the GloVe-based CNN-LSTM and HeBiLSTM models.

The 3.21% training and 22.07% validation loss of the ConvBidirectional-LSTM framework in 15 epochs are as shown in Figure 4.

Similarly, the 99.47% training and 93.25% validation accuracy of the proposed framework in 15 epochs are as shown in Figure 5.

The confusion matrix and the ROC curve of the proposed ConvBidirectional-LSTM framework are depicted in Figures 6 and 7.

Table 5 shows that Wen and Li [13] proposed three variants of the hybrid RNN and CNN model: Attention Recurrent Convolutional (ARC), Recurrent Convolutional (R.C.), and Multiple Attention Recurrent Convolutional (M_ARC) models. The evaluated performance accuracy of the ARC, R.C., and M_ARC models was 83.10%, 83.20%, and

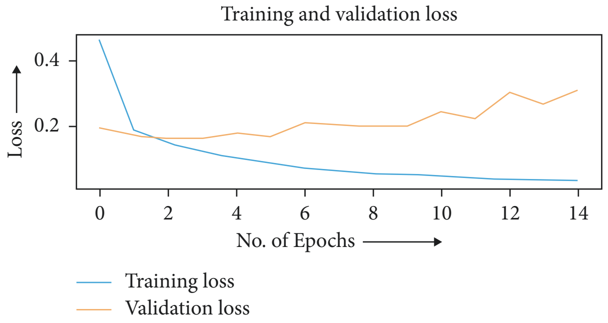

Figure 4: Training and validation loss function of ConvBidirectional-LSTM.

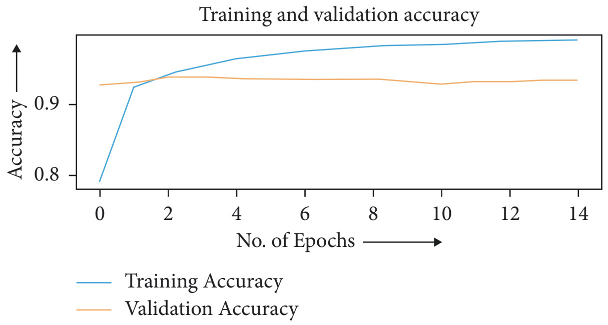

Figure 5: Training and validation accuracy function of ConvBidirectional-LSTM.

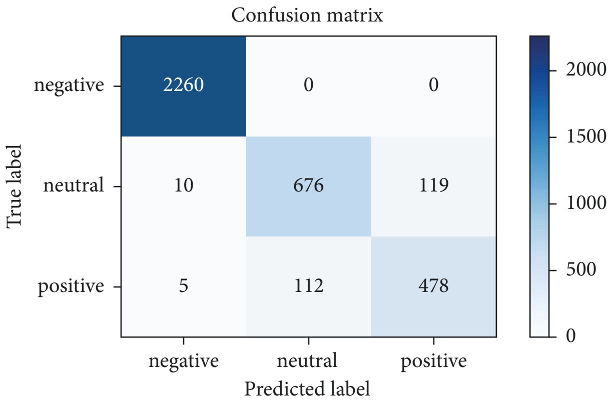

Figure 6: Confusion matrix of ConvBidirectional-LSTM.

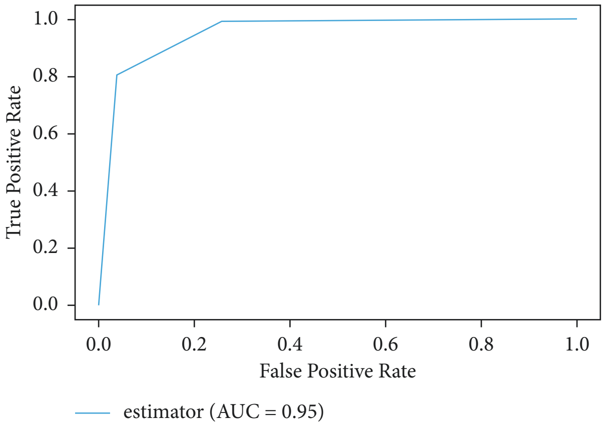

Figure 7: ROC curve of ConvBidirectional-LSTM.

Table 5: Accuracy comparison of the proposed model with the existing state-of-the-art models on the US airline dataset.

Authors	Model	Accuracy (%)
Wen and Li [13]	ARC	83.10
	RC	83.20
	M_ARC	83.30
Dang et al. [10]	TF-IDF-CNN	68.79
	Word embedding-CNN	82.36
	TF-IDF-RNN	61.74
	Word embedding-RNN	83.76
Umer et al. [12]	CNN-LSTM	82.00
Jain et al. [11]	LR	88.20
	NB	87.10
	DT	78.10
	SVM	80.20
	CNN	87.10
	LSTM	88.20
	CNN-LSTM	91.30
Basiri et al.[9]	ABCDM	92.75
Proposed model	GloVe-CNN-LSTM	92.79
	GloVe-HeBiLSTM	92.47
	ConvBidirectional-LSTM	93.25

83.30%, respectively, on the US airline dataset for positive, negative, and neutral sentiments. Jain et al. [11] proposed the CNN-LSTM model on the US airline dataset for classifying positive and negative opinions having a performance accuracy of 91.30% and compared the proposed CNN-LSTM model with other machine learning models. Basiri et al. [9] proposed the attention-based bidirectional CNN-RNN deep model (ABCDM) for analyzing positive and negative sentiments on the US airline sentiment dataset with a performance accuracy of 92.75%.

Table 5 illustrates the remarkable accuracy of 93.25% achieved by the proposed ConvBidirectional-LSTM framework on the US airline dataset for predicting positive, negative, and neutral sentiments. The overall result shows that the proposed ConvBidirectional-LSTM model is better than the existing state-of-the-art models.

6. Conclusion

The proposed sentiment analysis approach aimed to develop an efficient deep learning-based sentiment prediction model. This model analyzes the users' sentiments using textual data from social media platforms. The data for experimental evaluation were collected from the microblogging site Twitter. This research article suggested a ConvBidirectional-LSTM model for sentiment prediction. The sentiment prediction has experimented with three sentiment labels: positive, negative, and neutral. Although earlier works of literature only considered two, positive and negative, labels for their sentiment analysis, the performance of the proposed approach was evaluated by different performance metrics such as F_1 score, recall, accuracy, and precision. Further, the proposed ConvBidirectional-LSTM method was compared with the existing state-of-the-art models, GloVe-based CNN-LSTM and HeBiLSTM models.

Although this research work was carried out on an airline dataset that contains textual data in the English language, the authors are looking to build their proposed model to use other languages in future work. The architecture of the ConvBidirectional-LSTM framework could also be improved to enhance sentiment classification accuracy by implementing other hybrid deep learning models.

Data Availability

The dataset was taken from online digital libraries like Kaggle and CrowdFlower. The authors have included the link also in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] H. Sadr, M. M. Pedram, and M. Teshnehlab, "Multi-view deep network: a deep model based on learning features from heterogeneous neural networks for sentiment analysis," *IEEE Access*, vol. 8, pp. 86984–86997, 2020.

[2] A. Kumar and G. Garg, "Systematic literature review on context-based sentiment analysis in social multimedia," *Multimedia Tools and Applications*, vol. 79, no. 21-22, pp. 15349–15380, 2020.

[3] I. Guellil and K. Boukhalfa, "Social big data mining: a survey focused on opinion mining and sentiments analysis," in *Proceedings of the 2015 12th international symposium on programming and systems (ISPS)*, April 2015.

[4] M. Bouazizi and T. Ohtsuki, "Multi-class sentiment analysis on twitter: classification performance and challenges," *Big Data Mining and Analytics*, vol. 2, no. 3, pp. 181–194, 2019.

[5] G. A. Ruz, P. A. Henríquez, and A. Mascareño, "Sentiment analysis of twitter data during critical events through bayesian networks classifiers," *Future Generation Computer Systems*, vol. 106, pp. 92–104, 2020.

[6] B. Jang, M. Kim, G. Harerimana, S. u. Kang, and J. W. Kim, "Bi-lstm model to increase accuracy in text classification:

combining word2vec cnn and attention mechanism," *Applied Sciences*, vol. 10, no. 17, p. 5841, 2020.

[7] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: a survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.

[8] D. Mahto and S. Nair, "A real time vision based smart bulb using image processing," *International Journal of Computer Application*, vol. 164, no. 11, pp. 46–54, 2017.

[9] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "Abcdm: an attention-based bidirectional cnn-rnn deep model for sentiment analysis," *Future Generation Computer Systems*, vol. 115, pp. 279–294, 2021.

[10] N. C. Dang, M. N. Moreno-García, and F. D. L Prieta, "Sentiment analysis based on deep learning: a comparative study," *Electronics*, vol. 9, no. 3, p. 483, 2020.

[11] P. K. Jain, V. Saravanan, and R. Pamula, "A hybrid cnn-lstm: a deep learning approach for consumer sentiment analysis using qualitative user-generated contents," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 20, no. 5, pp. 1–15, 2021.

[12] M. Umer, I. Ashraf, A. Mehmood, S. Kumari, S. Ullah, and G. S. Choi, "Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model," *Computational Intelligence*, vol. 37, no. 1, pp. 409–434, 2021.

[13] S. Wen and J. Li, "Recurrent convolutional neural network with attention for twitter and yelp sentiment classification: arc model for sentiment classification," in *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 1–7, NY, U.S.A, December 2018.

[14] D. Mahto and S. C. Yadav, "Emotion analysis from text data using hebilstm," *The Journal of Oriental Research Madras*, vol. 92, no. 4, pp. 169–178, 2021.

[15] D. Mahto and S. C. Yadav, "Hierarchical bi-lstm based emotion analysis of textual data," *Bulletin of the Polish Academy of Sciences, Technical Sciences*, vol. 70, 2022.

[16] I. Chaturvedi, E. Cambria, R. E. Welsch, and F. Herrera, "Distinguishing between facts and opinions for sentiment analysis: survey and challenges," *Information Fusion*, vol. 44, pp. 65–77, 2018.

[17] D. Mahto, S. Nair, and S. Nair, "A review on smart bulb & proposed a real time vision based smart bulb using image processing," *International Journal of Engineering Trends and Technology*, vol. 44, no. 4, pp. 195–201, 2017.

[18] C. F. Hofacker, E. C. Malthouse, and F. Sultan, "Big data and consumer behavior: imminent opportunities," *Journal of Consumer Marketing*, vol. 33, no. 2, pp. 89–97, 2016.

[19] E. Cambria, D. Das, S. Bandyopadhyay, and A. Feraco, "Affective computing and sentiment analysis," in *A practical guide to sentiment analysis*, pp. 1–10, Springer, Berlin. Germany, 2017.

[20] X. Q. Liu, Q. L. Wu, and W. T. Pan, "Sentiment classification of micro-blog comments based on Randomforest algorithm," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 10, 2019.

[21] R. Monika, S. Deivalakshmi, and B. Janet, "Sentiment analysis of us airlines tweets using lstm/rnn," in *Proceedings of the 2019 IEEE 9th International Conference on Advanced Computing (IACC)*, pp. 92–95, IEEE, Tiruchirappalli, India, December 2019.

[22] F. Rustam, I. Ashraf, A. Mehmood, S. Ullah, and G. Choi, "Tweets classification on the base of sentiments for us airline companies," *Entropy*, vol. 21, no. 11, p. 1078, 2019.

[23] S. Kumar and M. Zymbler, "A machine learning approach to analyze customer satisfaction from airline tweets," *Journal of Big Data*, vol. 6, no. 1, p. 62, 2019.

[24] H. Hakh, I. Aljarah, and B. Al-Shboul, "Online social media-based sentiment analysis for us airline companies," *New Trends in Information Technology*, vol. 176, 2017.

[25] J. Acosta, N. Lamaute, M. Luo, E. Finkelstein, and C. Andreea, "Sentiment analysis of twitter messages using word2vec," in *Proceedings of the Student-Faculty Research Day, CSIS*, Pace University, Granada, Spain, October 2017.

[26] T. R. Gadekallu, D. S. Rajput, M. P. K. Reddy et al., "A novel PCA-whale optimization-based deep neural network model for classification of tomato plant diseases using GPU," *Journal of Real-Time Image Processing*, vol. 18, no. 4, pp. 1383–1396, 2021.

[27] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, D. S. Rajput, R. Kaluri, and G. Srivastava, "Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis," *Evolutionary Intelligence*, vol. 13, no. 2, pp. 185–196, 2020.

[28] S. C. Yadav, P. Kumar, and V. Kumar, "Performance analysis of deep neural network algorithm with optimizer for rumour detection," *Int. J. of Aquatic Science*, vol. 12, no. 3, pp. 38–45, 2021.

[29] D. Pandey, V. K. Nassa, A. Jhamb et al., "An integration of keyless encryption, steganography, and artificial intelligence for the secure transmission of stego images," in *Multidisciplinary Approach to Modern Digital Steganography*, pp. 211–234, IGI Global, Pennsylvania, PA, U.S.A, 2021.

[30] P. Kumar and R. S. Thakur, "Diagnosis of liver disorder using fuzzy adaptive and neighbor weighted k-nn method for lft imbalanced data," in *Proceedings of the 2019 International Conference on Smart Structures and Systems (ICSSS)*, March 2019.

[31] P. Kumar and R. Singh Thakur, "An approach using fuzzy sets and boosting techniques to predict liver disease," *Computers, Materials & Continua*, vol. 68, no. 3, pp. 3513–3529, 2021.

[32] P. Kumar and R. S. Thakur, "Liver disorder detection using variable- neighbor weighted fuzzy K nearest neighbor approach," *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16515–16535, 2021.

[33] S. M. Basha and D. S. Rajput, "Parsing based sarcasm detection from literal language in tweets," *Recent Patents on Computer Sciences*, vol. 11, no. 1, pp. 62–69, 2018.

[34] S. M. Basha and D. S. Rajput, "A supervised aspect level sentiment model to predict overall sentiment on tweeter documents," *International Journal of Metadata Semantics and Ontologies*, vol. 13, no. 1, p. 33, 2018.

[35] S. M. Basha and D. S. Rajput, "An innovative topic-based customer complaints sentiment classification system," *International Journal of Business Innovation and Research*, vol. 20, no. 3, p. 375, 2019.

[36] E. Sezgen, K. J. Mason, and R. Mayer, "Voice of airline passenger: a text mining approach to understand customer satisfaction," *Journal of Air Transport Management*, vol. 77, pp. 65–74, 2019.

[37] P. Kanerva, J. Kristoferson, and A. Holst, "Random indexing of text samples for latent semantic analysis," *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 22, 2000.

[38] S. M. Basha and D. S. Rajput, "A roadmap towards implementing parallel aspect level sentiment analysis," *Multimedia Tools and Applications*, vol. 78, no. 20, pp. 29463–29492, 2019.

[39] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, "Sentiment analysis of comment texts based on bilstm," *IEEE Access*, vol. 7, pp. 51522–51532, 2019.

[40] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," 2014, https://arxiv.org/abs/1404.2188.

[41] Q. Qian, M. Huang, J. Lei, and X. Zhu, "Linguistically Regularized Lstms for Sentiment Classification," 2016, https://arxiv.org/abs/1611.03949.

[42] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Advances in Neural Information Processing Systems*, vol. 13, 2000.

[43] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[44] J. Pennington, R. Socher, and C. D. M. Glove, "Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, CA, U.S.A, January 2014.

[45] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[46] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[47] D. Li and J. Qian, "Text sentiment analysis based on long short-term memory," in *Proceedings of the 2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI)*, pp. 471–475, IEEE, Wuhan, China, October 2016.

[48] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," *Lecture Notes in Computer Science. In International Workshop on Artificial Neural Networks*, Springer, Berlin, Heidelberg, pp. 195–201, 1995.

[49] M. H. Su, C. H. Wu, K. Y. Huang, and Q. B. Hong, "Lstm-based text emotion recognition using semantic and emotional word vectors," in *Proceedings of the 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, May 2018.

[50] Z. Wang and L. Jia, "Short-term photovoltaic power generation prediction based on lightgbm-lstm model," in *Proceedings of the 2020 5th International Conference on Power and Renewable Energy (ICPRE)*, pp. 543–547, IEEE, Shanghai, China, September 2020.

[51] J. Du, Y. Cheng, Q. Zhou, J. Zhang, X. Zhang, and G. Li, "Power load forecasting using bilstm-attention," in *Proceedings of the IOP Conference Series: Earth and Environmental Science*, vol. 440, no. 3, March 2020, Article ID 032115.

[52] S. Tam, R. B. Said, and O. O. Tanriöver, "A convbilstm deep learning model-based approach for twitter sentiment classification," *IEEE Access*, vol. 9, pp. 41283–41293, 2021.

[53] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional cnn-lstm model," in *Proceedings of the 54th annual meeting of the association for computational linguistics*, vol. 2, pp. 225–230, Berlin, Germany, January 2016.

[54] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for Activation Functions," 2017, https://arxiv.org/abs/1710.05941.

[55] M. Gupta, R. Kumar, H. Walia, and G. Kaur, "Airlines based twitter sentiment analysis using deep learning," in *Proceedings of the 2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, October 2021.

[56] C. François, "Keras: the python Deep Learning Library," 2015, https://keras.io/.