

CoBiAt: A Sentiment Classification Model Using Hybrid ConvNet-Dual-LSTM with Attention Techniques

Roop Ranjan and A K Daniel

Department of Computer Science and Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, UP, India

Keywords: deep learning, Dual-LSTM, keras, FasText, attention, emotion analysis

Received: Januar 14, 2022

Many researchers have recently turned their attention to emotion analysis as a resultant to the number of social reviews of various services. User behaviour may be better understood with the plethora of data, which makes it possible to work toward enhancing QoS. The critical areas of research in language processing are text categorization, which places unorganised data into relevant categories. In several natural language processing (NLP) applications, LSTM and CNN are utilised for text classification. CNN models use techniques to obtain top-level features. In this study, an attention-based model using Dual-LSTM and ConvNet has been proposed. For effectiveness verification of the model, it is trained using two unique datasets. The proposed hybrid model has demonstrated a significant performance gain when compared to previous deep learning techniques. In comparison to other traditional machine learning models, the suggested approach yields outcomes with a higher level of accuracy.

Povzetek: Predstavljena je metoda CoBiAt za klasifikacijo čustev z uporabo Dual-LSTM in ConvNet .

1 Introduction

The widespread usage of social media allows people to provide comments on events, situations, services, and products qualities [1]. These comments are frequently based on user experiences, which may include good or bad thoughts about items or services. These suggestions will assist firms in improving their services, allowing them to generate enhanced profit. As a result, it is critical to assess user input gathered from social networking websites. Analysis of Sentiments is useful in expressing users' opinions (positive, neutral or negative) about their services through text data [2]. It has also been observed that scholars are becoming more interested in social media platforms such as Facebook and Twitter. Businesses embrace public opinion analysis because it describes human activity and behaviour, as well as how individuals are influenced by the viewpoints of others.

In recent era, the widespread applications using deep learning has led to advancements in image processing, natural language processing, and speech recognition. Deep learning is superior to machine learning in classification of sentiment analysis problems due to the availability of large datasets and the inexpensive mass fabrication of capable Graphics Processing Unit (GPU) units [5]. Deep learning is extensively utilized in the models based on natural language processing (NLP) [6], including emotion analysis, because to its autonomous learning characteristics. The two most often used deep learning algorithms in sentiment analysis of reviews are CNN and Recurrent Neural Network (RNN). Gradient disappearing and exploding problems plague RNN to a large extent. RNN is challenging to train for long distance

correlations for a given series because of these difficulties. The RNN model underpins BiLSTM, which has shown promising results in text-based sentiment analysis. An LSTM model like this features channels for two-way communication to help the network comprehend its environment. The forward and backward layers of BiLSTM allow the network to access the sequence's prior and subsequent contexts [7].

In the past few years, many CNN- or RNN-based methods for classifying text have been proposed [8, 9]. CNNs can learn the local behaviour from temporal data, but not sequential correlations. RNNs are specialised for sequential modelling, as opposed to CNNs, but are unable to extract features in parallel. Traditional RNNs, however, result in an exploding and vanishing state versus its gradient for extended data sequences. Vanishing gradient and gradient explosion problems are successfully solved by long short-term memory (LSTM) [10], a type of RNNs architecture with LSTM units as hidden units.

The attention-based mechanism aids in enhancing the performance of deep learning models, sentence summarization, and reading comprehension, as demonstrated in the machine learning translation process [11, 12]. The majority of deep learning implementations for text analysis use word embedding techniques to produce feature vectors from the dataset. BiLSTM is unable to prioritise the critical information from the data and just collects contextual information from the features. In contrast to BiLSTM, CNN has a convolutional layer that extracts and shrinks vector features.

In this research we aim to construct a novel text classification utilising a hybrid deep learning attention-oriented optimal model that integrates the strong

characteristics of CNN with BiLSTM employing an attention mechanism in order to handle the aforementioned problem. By adding a convolutional layer to the CNN model with attention features, the attention-based Conv-BiLSTM mechanism, a unique approach, aims to solve the shortcomings of BiLSTM. The suggested model's overall methodology involves training the input data using the Keras skip-gram model, which is then passed to the Convolution Layer, which draws out the data's basic semantic information. The BiLSTM layer, which combines the attention-based approach to identify which characteristics are significantly associated with semantics and should be employed for final classification, receives the feature vectors obtained by the Conv layer.

The training and performance evaluation of the proposed model is performed on two datasets; the first set of data is the collection of tweets for Indian Railways (hereafter IR Dataset) starting from 01st October 2019 and 31st October 2019 [13] and the second dataset is IMDB reviews. Word embedding is performed using two prominent word embedding algorithms, FasText and Keras embedding. Experiments demonstrated that the Self Attention Based Conv-BiLSTM model outperformed other deep learning models and conventional machine learning methodologies.

The following are the key contributions of the research:

(1) Two distinct Word embedding approaches FasText and Keras embedding were used to render tweets as word vectors. Both of the strategies for word embedding make use of pre-trained, supervised word vectors that are able to capture the semantics of individual words and are taught using a large corpus of words. The effectiveness of the proposed CoBiAt model will be evaluated through the utilisation of these two-word vector models as the objective.

(2) ConvNet that has been coupled with BiLSTM and the Self Attention methodology has been offered for classifying the reviews. The ConvNet module collects local features via word embedding, the Self Attention-based BiLSTM module extracts long-distance associations, and the selected features are then categorised in the classification result.

(3) The experimental results are compared with other popular deep learning-based techniques and standard machine learning approaches to prove the potential of the proposed optimised model.

There are several drawbacks in deep learning models like CNN, RNN and LSTM etc. The main issue with CNN is that it does not provide clear encoding the orientation and position of content. Whereas RNN suffers from exploding problems and gradient vanishing issues. The LSTM technique takes longer time for training the dataset also implementing dropout in LSTM is a very tedious task. All the above issues with deep learning methods motivated us to design an approach that overcomes the drawbacks of deep learning methods. Therefore, we performed regressive experiments and proposed a hybridized model that combines the strong features of CNN with BiLSTM using an efficient attention-

mechanism to further optimise the performance and accuracy of the classification provided by the model.

The following is the structure of this research paper: The background and important research for text sentiment categorization using ConvNet and attention-based Dual-LSTM are discussed in Section 2. Section 3 describes in detail how the proposed model works. Section 4 describes the environment setup for executing the CoBiAt model. Section 5 discusses in depth the experimental outcomes for the research and comparison with other models. Section 6 finishes with a conclusion and suggestions for future research.

2 Related work

In recent times, public opinion analysis on network Deep learning techniques have shown remarkable accomplishments in the area of natural language processing in recent years. Deep learning is subfield of machine learning that seeks to represent high-level of abstractions in the given data. This is accomplished through the use of model architectures with complex structures or those built of several nonlinear transformations [14]. Convolutional Neural Network (CNN) learns complicated, high-dimensional, and non-linear mapping relationships by fully utilising the structure of multi-layer perceptron. It has been frequently utilised and has produced good results in image identification and speech recognition applications [15].

CNN was proposed for use in natural language processing by [16], who also constructed a dynamic Convolution Neural Network (DCNN) technique to analyse dataset of varying lengths. In [17], the authors developed a method for analysing opinions regarding health services. They amassed 2,026 tweets using Twitter hashtags to compile their dataset. They compared DNN and CNN with word2vec embeddings as two DL models. CNN's model was the most accurate. The model was trained on a fairly limited dataset in this work, and neither model addressed the negation problem. In [18], five different combinations of LSTM models were utilised to analyse tweets. To train the models, they utilised both dynamic and static CBOW and a word embedding. The results demonstrated that the integrated LSTM model trained using dynamic CBOW performed better than the other models.

Many neural network-based techniques have recently been demonstrated to be effective in a number of sentimental analysis applications. Given the attention-based mechanism's tremendous success in natural language processing, its application in Sentiment Analysis tasks has gained in prominence. The paper [19] proposed using the deep attention mechanism to analyse user evaluations and produced better results than a recurrent neural network (RNN). The model [20] suggested a bidirectional gated recurrent unit-based position-aware bidirectional attention network (PBAN) (bi-GRU). To tackle this difficulty, the LSTM model [20] was suggested, which had the capacity to maintain sequence information and produced good results on several sequence modeling tasks.

A powerful tool for capturing the relationship between context and aspect at the next level is an attention mechanism. The authors in [21] made a multi-grained attention network (MGAN) that uses both fine-grained and coarse-grained attention mechanisms to collect information about how aspect and context interact with one another. In another research [22] used the concept of feed-forward networks and multi-head attention (MHA) to effectively extract the hidden context presentation of and embeddings of aspects. In the paper [23], movie reviews from IMDb were analysed to see how people felt about them. Some pre-processing steps were taken to get rid of characters, symbols, words that were repeated, and "stop words." Then, CountVectorizer was used to extract features. The authors proposed CNN model and compared with several traditional models. The proposed CNN model achieved 99.33% accuracy. The dataset has 3000 reviews that are either good or bad. Meng et al. [24] used a CNN to get the higher-level feature representation from a better layer of word embedding. After a BiLSTM takes in local and global semantic information, an attention layer is used to focus on important term characteristics of aspects. Furthermore, Ma et al. created an attention-based LSTM that uses the common-sense knowledge of sentiment-related concepts proposed in SenticNet for incorporating external knowledge [25].

In contrast to conventional machine learning classification algorithms, LSTM has demonstrated its effectiveness in achieving high classification accuracy [26]. This article [27] implemented deep CNN and BiLSTM. Deep CNN was implemented on character-level embeddings for improving word embeddings' information. Then, bidirectional LSTM is used for classifying the sentences according to their sentiment. This article emphasises data standardisation in order to obtain high performance, the researchers have created a tweet processor model to remove unnecessary terms from tweets. In their research [28], offer Hybrid two Convolutional Neural Networks and Bidirectional LSTM, a further variant of hybridized deep learning architecture for sentiment classification. Two CNN layers and a bidirectional LSTM layer are employed here. Seven datasets are evaluated using three pre-trained word vectors, GloVe, Word2Vec, and FasText. Word2Vec has been observed to be more efficient than the other two-word vectors.

In another research [29], the authors suggested a network of deep memory for sentiment classification. The proposed method may simultaneously record both user and product information. Inference-based memory components and a large long-term memory that also serves as a knowledge basis compose this memory network. Model architecture has two parts. Each document is represented by an LSTM, and its rating is predicted using a deep memory network with several levels (hops), all of which are content-based attention models.

The researchers in [30] used Dual LSTM and Keras word embedding to classify traveller attitudes. The authors' proposed model performed word embedding using the two distinct words embedding algorithms

word2Vec and Keras embedding. Keras embedding provided better results as compared to that of word2Vec. The authors in [31] devised hierarchy-based attention (HA) technique for capturing the hierarchical structure of documents at the sentence and text levels, where information of varying significance was given special treatment while generating document representations. Because most prior approaches focused just on localized information of contents and ignored preferences of global users and quality of product, [32] proposed a model for classification of sentiment using attention method for product information for global users. For text sentiment analysis, [33] recommended combining CNN with three distinct attention mechanisms: LSTM attention, vector attention, and pooling attention. [34] used a self-attention with a sparse technique for determining text emotion polarity by capturing the significance of each word. In another work, [35] looked at the challenge of classifying personality traits based on textual content. The authors used a hybrid CNN+ LSTM model to classify the text in a different personality trait. The research performed by the author's shows that CNN is a powerful method for selecting the best characteristics that improve prediction accuracy and the LSTM model keeps earlier context information, which makes it easier to use important context information at the start of a phrase. Their proposed model proved to be significantly superior to other models. However, the lack of attention model was observed in their results.

In innovative research in field of bio-medical engineering in [36], researchers presented a data analytical system for EEG utilising a multi-layer Gated Recurrent Unit (GRU) for anomaly identification. The proposed model consists of four stages from data collection to model performance evaluation. Using a publicly accessible EEG dataset, the suggested model achieved accuracy of 96.91 percent, sensitivity of 97.95 percent, specificity of 96.16 percent, and 96.39 percent F1 score. In [37] Wang et al., demonstrated the concept of weak tagging for sentiment classification. The authors presented a BiLSTM emotion categorization model with multi stages of training and a emotion classification model with weak tagging data denoising. The model for emotion classification based on weak type tagging information denoising had the best classification performance of all the experimental groups, but the model was also observed to be the most time-consuming.

In the field of medical imaging deep learning has shown promising results. In [38], authors have demonstrated a model based on Transfer Learning (TL). Using a state-of-the-art CNN for fundus image processing, we customised TL for mild and multi-class DED (diabetic eye disease) cases in this study. Using fine-tune, optimization the researchers achieved 88.3% accuracy. In another article [39], a unique hybrid CNN-BiLSTM deep learning strategy for four-level facial pain recognition is proposed. To achieve the results of pain intensity estimation satisfactorily, the fully connected layer of the VGG-Face was optimized for this task by addition of a fully connected layer, and the dimensions of the extracted features was reduced using PCA (Principal Component

Analysis) to improve the algorithm's overall computational efficiency. The improved algorithm achieved an AUC of 98.4% and a test precision of 90%. By training a large number of sentiment text corpora, Using algorithms convex hull and convolution neural network, [40] proposes a model for fault detection in wireless sensor networks. The authors performed several experiments and found that CNN with Naïve Byes is proving to be better and more efficient. The authors in [41], presented a model using block-chain with deep learning. Described all the fundamental ideas involved in the management and security of such data, and offered a novel solution to handle the hospital's big data utilising Deep Learning and Block-Chain technology to ensure their safety.

Since RNN can preserve a sequence of information over time, it is a helpful supplement to CNN; nonetheless, RNN is severely affected by the Gradient explosion problem described by [42]. Because of these issues, distance correlation in a sequence is difficult to train with RNN. Bi-LSTM is an RNN model that has reportedly shown promising outcomes in the analysis of sentiment in the given text. It contains two LSTMs to allow the network for better understanding of the contexts provided. Backward LSTM's and forward layers grant the for accessing the sequence's preceding and subsequent context. Text sentiment classification, on the other hand, uses vectors to represent the text, which is often performed in a large-dimension space. When the Bi-LSTM retrieves relevant knowledge from several obtained features, it is unable to place a premium on the most important data [43].

Considering the above issues in deep learning methods for text classification, the CoBiAt model has been proposed here that combines the strong feature of CNN and Bi-LSTM with attention model. The performance of the model indicates that the proposed model has the potential to tackle above issues in deep learning models.

3 Proposed method

The proposed attention-based method has the following layers:

- 3.1 Input and Pre-Processing
- 3.2 Word Vector Matrix
- 3.3 ConvNet Layer
- 3.4 Dual-LSTM Layer
- 3.5 Attention Layer

3.1 Input and Pre-Processing Layer

In this part, the words are standardized and cleansed by changing them from a human language to text format in order to eliminate superfluous elements. This phase assists classifiers in achieving high performance and rapid sentiment classification. In this work, steps included tokenization of sentences into related words by using

NLTK (Natural Language Toolkit), a popular python library: convert upper case to lower case; duplicate text removal, removal of special characters, reduction of several spaces, hashtags, URL, mention, punctuations and other stopwords; and lemmatization of texts in dataset.

3.2 Word vector matrix layer

The Word Vector Matrix layer embeds pre-processed input tokens. Token representation reveals hidden ties between words that often appear together. The input dataset was trained using the Keras Skip-gram model. Here the text from the input dataset has N-words, vector of the t^{th} words is represented by w_t with $t \in [1, T]$ in the given text. Here the text is given with words w_t , the words are embedded by layers for vectors using a layer of embedding W_p . x_t is depiction of vector for w_t , which is represented using Eq.1.

$$x_t = W_p w_t \quad (1)$$

The model was trained with skip-gram approach by optimising the average likelihood log. The method trains semantic embedding by prediction of the target word based on context and detecting semantic links between words.

3.3 ConvNet layer

With the ConvNet layer, the task of selecting features from input data is accomplished. In view of reducing the number of dimensions in the source text, ConvNet layers are employed. Several 1-Dimensional Convnet kernels are employed for conducting convolution for the input vectors in this study. Equation (2) creates a sequential vector of text by integrating the component vectors of word embedding:

$$X_{1:V} = [x_1, x_2, x_3, x_4, \dots, x_V], \quad (2)$$

where V represents the total size of tokens in the given text. Convolution kernels of different sizes are implemented on $X_{1:V}$ for capturing the n-gram characteristics (where $n=1,2,3$) of the given text using a one-dimensional Convolution to capture the intrinsic features. When a window of r words spanning from $v: v+r$ is input during the v^{th} convolution, the convnet process creates features set for such window as below:

$$h_{r,v} = \tan h(W_r x_{v:v+r-1} + d_r), \quad (3)$$

Where $X_{v:v+r-1}$, are the embedded vectors obtained of the given word in a given window, W_r is the weights matrix that can learn, and d_r is the bias used in embedding. Since every filter primarily has to be implemented to different parts of the word, therefore the filter's feature maps with the size of convolution r are:

$$h_r = [h_{r1}, h_{r2}, h_{r3}, h_{r4}, \dots, x_{V-r+1}]. \quad (4)$$

It is advantageous to utilise Convnet kernels of varying sizes in order to capture hidden connections between adjacent words. The most essential objective of

using a CNN for text-based feature retrieval is that it minimises the number of learnable parameters throughout the max-pooling-based process of feature learning. Multiple channels of convolution act on the input, with each channel holding data at distinct timestamps. Consequently, the output of each ConvNet channel during the max-pooling operation is the largest value of all timestamps for that channel. Max pooling is implemented here to the maps of features with convolution size r for each convolution kernel to produce:

$$s_r = \text{Max}^v(h_{r1}, h_{r2}, h_{r3}, h_{r4}, \dots, x_{V-r+1}) \quad (5)$$

For obtaining the final feature map of the window, p_r is combined for every filter size $r = 1, 2, 3$ and for extracting the n -gram (where $n=1, 2, 3$) hidden features:

$$h_r = [s_1, s_2, s_3]. \quad (6)$$

3.4 BiLSTM layer

The Dual-LSTM layer receives attributes as input from the ConvNet layer and extracts the final hidden state to produce features. The prior and subsequent context information is accessible to the Dual-LSTM module, and the data collected by the BiLSTM can be seen in two distinct text-based formats. The ConvNet feature sets are input into the Dual-LSTM model, which generates a sequential representation. By aggregating information for words in both directions (ahead and backward), Dual-LSTM obtains word annotations and therefore the annotations include contextual information. The forward LSTM represented as (\vec{L}_1) reads the sequences of features from first to last, whereas the reverse LSTM represented as (\vec{L}_2) reads from last to first. The word annotation received from \vec{L}_1 is represented by $L_{forward}$ and from \vec{L}_2 the annotation is represented by $L_{backward}$ the bidirectional process is iterated and the final feature representation output (L) is obtained as follows:

$$L = \{L_{forward}, L_{backward}\} \quad (7)$$

3.5 Attention layer

This output representation from the Dual-LSTM layer is delivered to the attention layer, which assesses which features are highly interconnected and should be used for final categorization. The attention mechanism, which is a fully linked layer with a softmax function that focuses on qualities of selected words to reduce the impact of less significant words on the text's sentiment, is a completely connected layer. The process of the attention layer is as follows:

The word annotation $L_{forward}$ is initially supplied to get $\vec{U}_{forward}$ by single perceptron as an uncovered format of $L_{forward}$. $\vec{U}_{forward}$ is represented as follows:

$$\vec{U}_{forward} = \tanh(f * L_{forward} + b) \quad (8)$$

Where weight and bias in the neuron is represented by f and b respectively. Hyperbolic tangent function is represented by $\tanh()$. The layer calculates the significance of each word based on the similarity between $\vec{U}_{forward}$ and a text level context vector $\vec{C}_{forward}$ for measuring the importance of every text. The softmax function is then utilized by the layer for calculating the normalized weight \tilde{Z}_{fwd} of each word as follows:

$$\tilde{Z}_{fwd} = \frac{\exp(\vec{U}_{forward} * \vec{C}_{forward})}{\sum_{i=1}^M \exp(\vec{U}_{forward} * \vec{C}_{forward})} \quad (9)$$

Here, the total number of texts in a particular set of texts is denoted by M .

The text level context vector $(\vec{C}_{forward})$ is an illustration of high level for the descriptive words for the set of word sequences that is initialized on a random basis and learned together throughout the phase of training.

The forward context representation Fc is then produced as a weight-based addition of the read word annotations in the forward direction on the basis of weight parameter \tilde{Z}_{fwd} . Fc is a component of the attention layer's output that may be represented as:

$$Fc = \sum(\tilde{Z}_{fwd} * L_{forward}) \quad (10)$$

Similar to \tilde{Z}_{fwd} , \tilde{Z}_{bwd} is calculated with the help of the backward direction hidden state $L_{backward}$. Hc , like Fc , is a backward context representation that is part of the attention layer's output, and it is represented as:

$$Hc = \sum(\tilde{Z}_{bwd} * L_{backward}) \quad (11)$$

The forward context representation Fc is concatenated with the backward context representation denoted as Hc , BiLSTM gets an interpretation for a particular sequence of features, and finally delivers the classification results. The attention layer improves the accuracy of prediction and decreases the size of learned weights required for predicting by using this method.

4 Experiments

4.1 Dataset

Experiments are carried out in this part to evaluate the performance of the proposed model for text categorization on two unique datasets. Using Roop

Ranjan et al. [30] as a starting point, we have 25000 tweets from people who used Indian Railways services on various days in October 2019. Table 1 breaks down these tweets into three different categories: positive, neutral, and negative. The binary-labeled set of IMDB movie reviews is the second dataset used. There are 14641 review tweets collected in the dataset as shown

in Table 2. The dataset from IMDB movie was also utilised to compare the model to prior sentiment categorization research. The IMDB movie dataset can be obtained from <https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews/metadata>.

In this section, experiments are conducted for

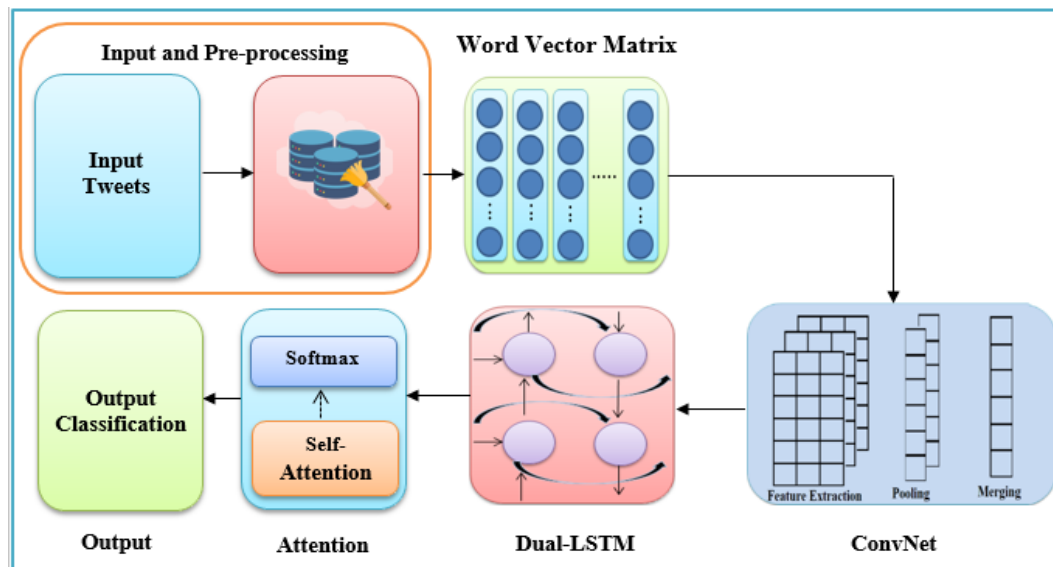


Figure 1: Proposed architecture

Table 1: Categorization details of tweets dataset

Tweets Dataset			
Size	Positive	Negative	Neutral
25000	10695	8953	5352

Table 2: Categorization details of IMDB movie reviews dataset

IMDB Movies Review Dataset			
Size	Positive	Negative	Neutral
50000	18324	20625	11051

Further both dataset are divided into Training, Validation and Testing Set using Gaussian distribution. The ratio of the total dataset is 60:20:20 for training, validation and testing dataset.

4.2. Experimental setup

Because of the GPU environment's support, Google Colab with Keras is being used with backend such as Tensorflow for Keras. Computing-intensive machine learning methods can be trained in shorter amounts of time when running on GPUs. In a GPU context, greater computational power is available, allowing for more training iterations while fine-tuning the machine learning models.

4.3 Setting of hyper-parameters

Implementing hyper-parameter tuning is crucial for High model performance can only be achieved if hyper-

parameter adjustment is implemented. The randomised search method is utilized to optimise hyper-parameters and improve the accuracy. Using a random combination of hyperparameters, randomised search determines the optimal answer for developing the model. Due to grid search's inability to perform well when there are a large number of dimensions, random search is preferred over grid search. Table 3 represents the hyper-parameters values using randomised search in the proposed model.

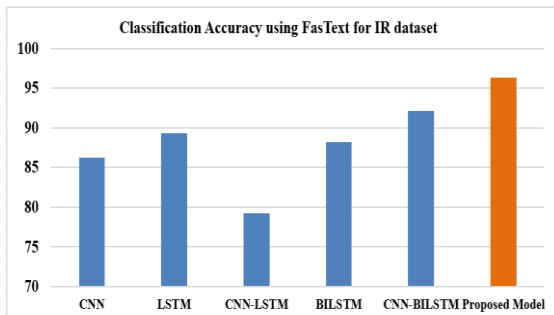
Table 3: Setting of hyper-parameters

Parameters	Values
Dimension(Embedding)	Keras(300)
Size of Kernel	5
Output Size(Dual-LSTM)	32
Filter Size	32
Function (Regularization)	L2
Activation	SoftMax
Weight Constraints	Kernel Constraints (max norm is 3)
Epochs Count	100
Batch Size	32
Batch Normalization	Yes
Learning Rate (LR)	0.03
Optimization	Adam

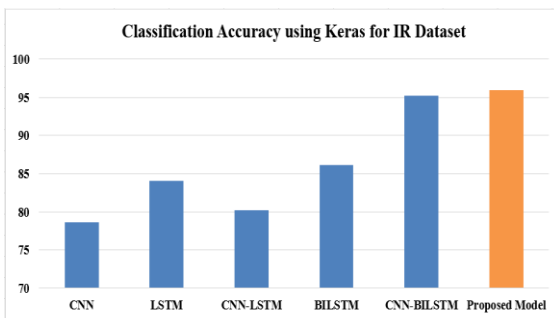
5 Results and discussion

5.1 Performance comparison

Using optimal hyper parameter values, the proposed model was compared to CNN, LSTM, CNN-LSTM, and BiLSTM, which are all deep learning-based models. The IR dataset and the IMDB Movies review dataset were used to make the comparison. Figure 2a shows the results of comparing the proposed model's overall accuracy with other deep learning models using FasText Embedding. On the IR dataset, CNN, LSTM, and BiLSTM all did better with FasText embedding than with Keras embedding. However, CNN-LSTM and CNN-BiLSTM with Keras embedding (Fig. 2b) gave more accurate results. The fact that the observation was made shows that the proposed model is much better than other methods. For the IR dataset, the proposed model is 96.32 percent accurate with FasText and 95.98 percent accurate with Keras.

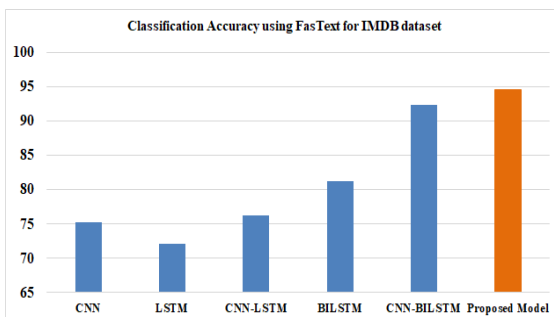


(a)

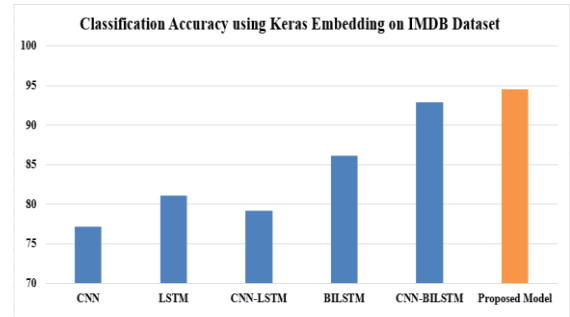


(b)

Figure 2: Accuracy of different models for IR dataset



(a)



(b)

Figure 3: Accuracy of different models based for IMDB dataset

The figure 3a and 3b provides comparison of the overall accuracy of the proposed architecture with other deep learning methods using Keras Embedding. The observation depicts that FasText embedding performs better than Keras embedding for IMDB dataset with an improvement of 1.12%.

5.2 Evaluation of performance

The proposed system's performance is assessed using the standard evaluation matrix illustrated in Figure 4.

		Predicted Class	
		No	Yes
Observed Class	No	TN	FP
	Yes	FN	TP

Figure 4: Standard evaluation parameters

The standard validation parameters are described as below:

- True Negative (TN) - These are accurately forecasted negative outcomes, demonstrating the value of actual class is zero and the outcome of the anticipated class is zero i.e. correct prediction of negative classes.
- True Positive (TP) - TP are observed positives that are accurately predicted and indicate that the outcome of the actual class and the outcomes of the expected class are positive i.e. correct prediction of positive classes.

False negative and false positives happen if actual class is different from the anticipated class.

- False Positive (FP) – observations of the anticipated class is positive and actual class is negative i.e. incorrect prediction of positive classes.
- False Negative (FN) - When the actual class is positive while the projected class is negative i.e. incorrect prediction of negative classes.

Using these standard parameters following rules are implemented for evaluation of effectiveness of the proposed hybrid model:

Precision (P) = Precision is termed as the proportion of correct anticipated positive outcomes to total projected positive outcomes.

$$P = \frac{TP}{TP + FP} \tag{12}$$

Recall(R) = the proportion of properly forecasted positive outcomes to the overall observations in the positive class.

$$R = \frac{TP}{TP + FN} \tag{13}$$

F-Measure (F) = the average of Recall and Precision is termed as F-Measure. Resulting in score takes into account both false negatives and false positives.

$$F = \frac{2 * (P * R)}{(P + R)} \tag{14}$$

Accuracy (A) = the most important performance parameter is accuracy; this is just the proportion of predicted observations which are correct to all observations.

$$A = \frac{TN + TP}{TN + TP + FN + FP} \tag{15}$$

Figure 5 illustrates the overall performance of two independent datasets and two distinct word embedding procedures utilising a variety of deep learning techniques. The suggested model outperformed competing strategies for both the IR and IMDB datasets. The overall precision of IR dataset with FasText embedding was observed 96.32% which is 3.07% more than the CNN-BiLSTM Model and outperformed three other models with a huge improvement. The recall value is improved by 2.12% than the nearest best performing model BiLSTM. The overall performance of the proposed model having F-measure was observed to be 96.16% and accuracy of 96.32% for FasText embedding. The performance of the proposed model has shown lesser performance when Keras Word Embedding is implemented on the different deep learning techniques and the proposed model. The model proposed here with Keras embedding displays the promising improvement for the classification with 96.01% precision, 96.32% recall, 96.16% F-measure and 95.98% accuracy. This was observed because CNN and LSTM lacked proper information about the forthcoming context of the network's huge corpus of words.

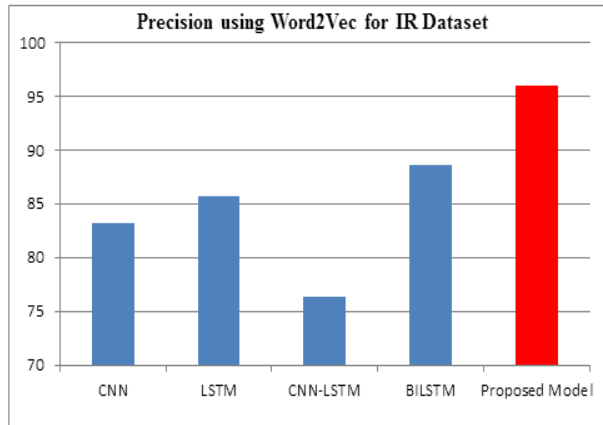
For IMDB dataset the study presented that the proposed model has shown improvement on other techniques. The proposed model with FasText shows the improvement of 3.03% over the CNN-BiLSTM model in predicting positive class prediction. As far as recall is concerned the improvement is 3.06%. For F-measure it shows improvement by 2.56%. The performance of proposed model over other studied techniques with Keras Embedding is also impressive with improvement of 94.63% in precision and 94.56% in accuracy.

Models	Dataset	FasText(%)				Keras(%)			
		P	R	F	A	P	R	F	A
CNN	IR	84.25	82.35	83.29	86.21	79.58	77.63	78.59	78.62
LSTM		86.98	85.25	86.11	89.32	81.52	82.32	81.92	84.12
CNN-LSTM		81.25	78.63	79.92	79.25	79.56	79.65	79.60	80.18
BILSTM		89.92	86.25	88.05	88.25	86.62	85.14	85.87	86.14
CNN-BILSTM		93.25	93.89	93.57	92.15	94.34	96.7	95.51	95.19
Proposed Model		96.32	96.01	96.16	96.32	96.01	96.32	96.16	95.98
CNN	IMDB Movie Reviews	74.21	75.21	76.32	75.21	76.33	77.1	76.71	77.17
LSTM		73.25	72.25	74.26	72.11	81.52	82.32	81.92	81.13
CNN-LSTM		76.32	79.35	79.31	76.32	79.56	79.65	79.60	79.16
BILSTM		82.26	77.65	82.15	81.25	86.62	85.14	85.87	86.14
CNN-BILSTM		90.36	91.62	92.65	92.35	92.63	93.42	93.02	92.9
Proposed Model		95.12	94.68	95.21	95.68	94.63	94.12	94.37	94.56

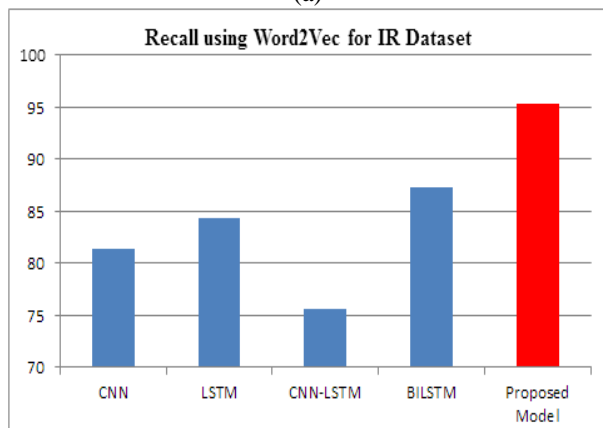
Figure 5: Performance evaluation on faxtext and keras word embedding

For US Airlines dataset the study presented that the proposed model has shown improvement on other techniques. The proposed model with Word2Vec shows an improvement of 7.34% over the BiLSTM model in predicting positive class prediction.

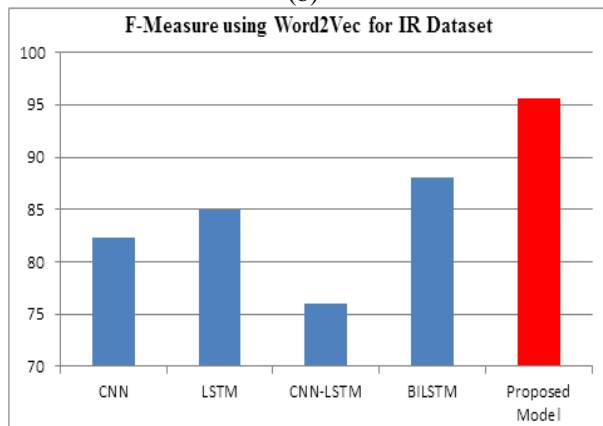
As far as the recall is concerned the improvement is 7.96%. For F-measure, it shows an improvement of 7.65%. The performance of the proposed model over other studied techniques with Keras Embedding is also impressive with an improvement of 9.39% in precision and 9.84% in accuracy.



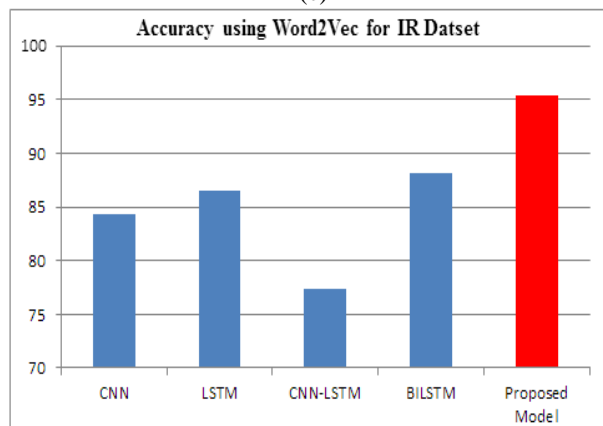
(a)



(b)



(c)

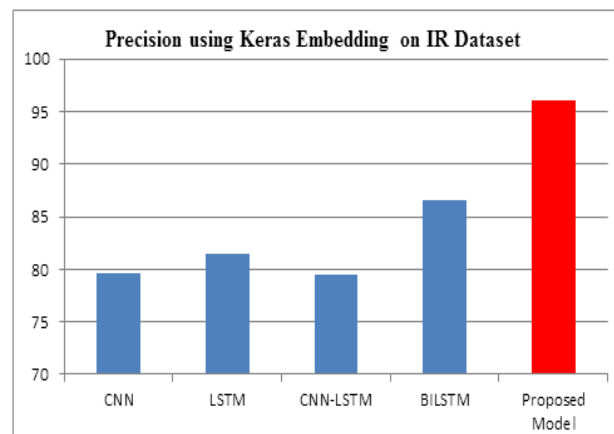


(d)

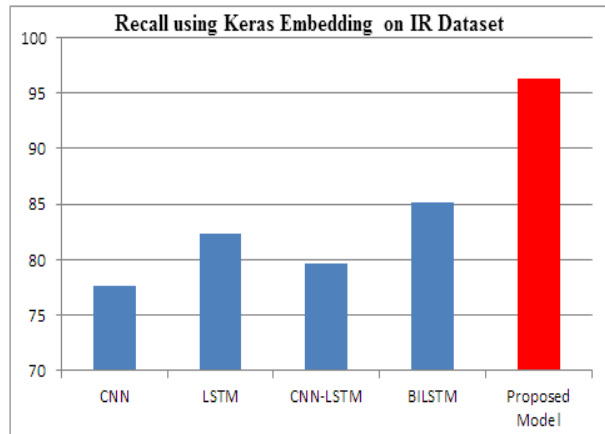
Figure 6: Model performance Word2Vec word embedding for IR dataset

Figure 6 shows the performance of different evaluation parameters using the proposed model on Word2Vec Embedding for the IR dataset. Experiments were performed on another deep learning model on the same dataset. Word2Vec effectively initializes word vectors for the IR datasets, as shown by the higher level of correctness of experimental outcomes. It is clear that the proposed self-attention-based classification model provides better results for all evaluation parameters as compared to other techniques with Word2Vec. The values of precision for CNN were observed at 83.16%, the LSTM model performed precision 85.65%, CNN-LSTM performed 76.32%. The BiLSTM deep learning model performed much better than the other three with the precision of 88.62% and 88.15% classification accuracy. The proposed Model outperformed other deep learning models with a precision of 95.96%. Other performance parameters are also evaluated for different models. Comparison of the proposed model is performed and the proposed self-attention-based model has shown an impressive improvement over other deep learning models with recall 95.32%, F-measure of 95.64%, and accuracy of 95.35%.

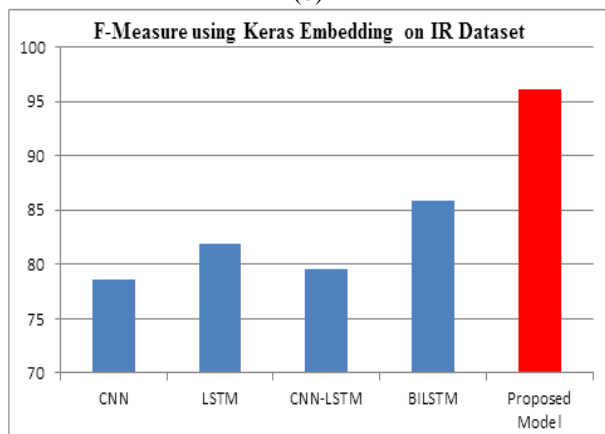
The accuracy of only CNN in the study was merely 82.30%, while the accuracy of experiments on BiLSTM was 87.99% on the IR dataset, indicating that utilizing CNN and BiLSTM separately to conduct sentiment analysis did not yield useful results. Further features of BiLSTM and CNN were combined to optimize the accuracy performance, this approach shows better efficiency than CNN and BiLSTM having accuracy of 91.60% on the IR dataset.



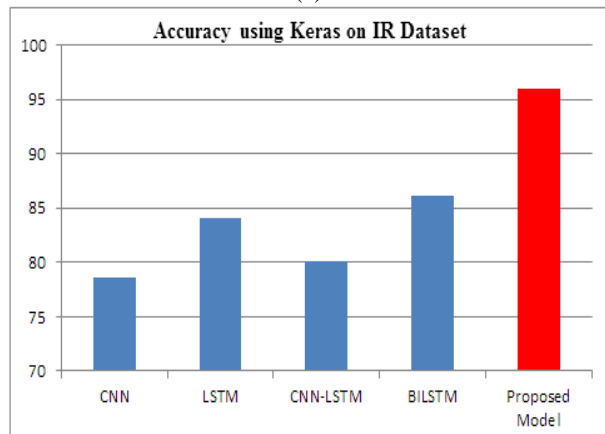
(a)



(b)



(c)

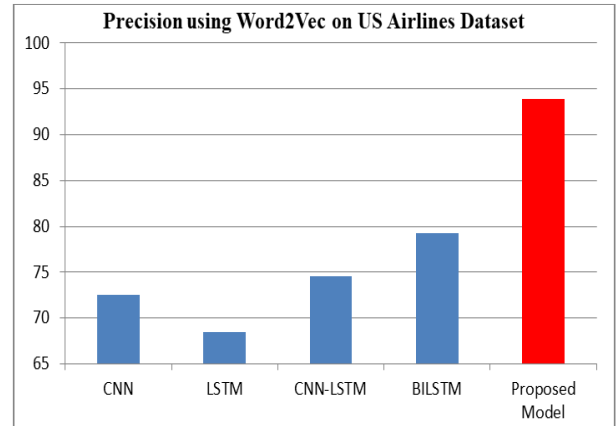


(d)

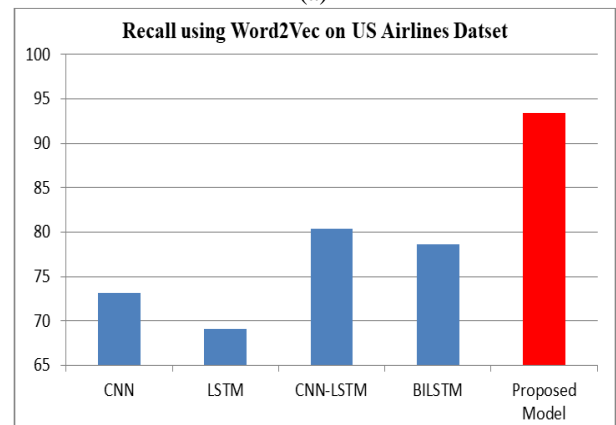
Figure 7: Model performance using Keras word embedding for IR dataset

Different deep learning models were also implemented on same IR dataset using Keras Embedding. Figure 7 displays the model performance on the said embedding. For each of the performance evaluation parameters the proposed model has shown significant improvement over others. The overall accuracy of the proposed optimized model was 9.84% better than BiLSTM method. Precision, Recall, and F-Measure were 9.39%, 11.18%, and 10.29% higher than the BiLSTM model. This indicates clearly that CNN and BiLSTM can't offer great results on their own since CNN can't learn the correlation sequence for long-term dependencies and BiLSTM can't extract local

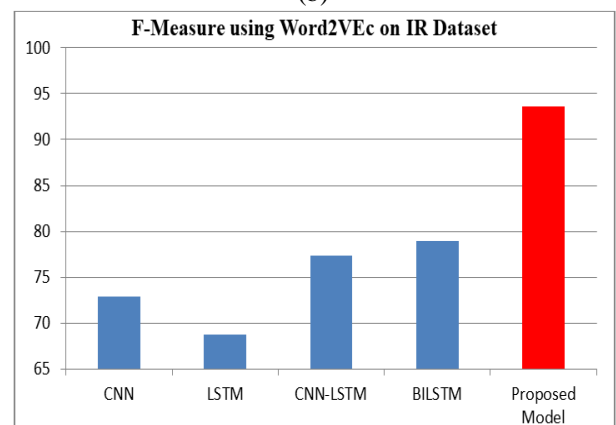
characteristics. The combination of CNN and BiLSTM is merged with self-attention; the model is able to learn each word in tweets more effectively since it contains enough word context information based on previous and future context.



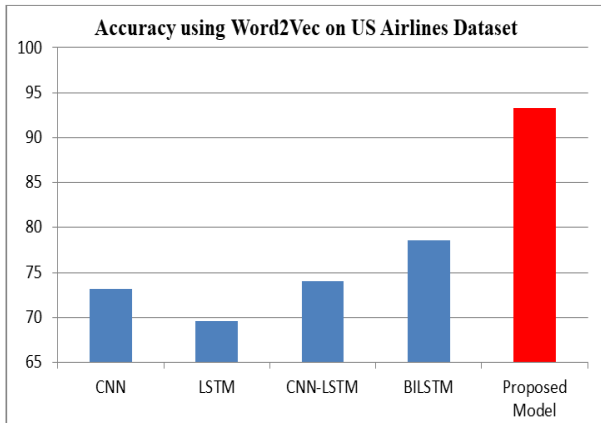
(a)



(b)



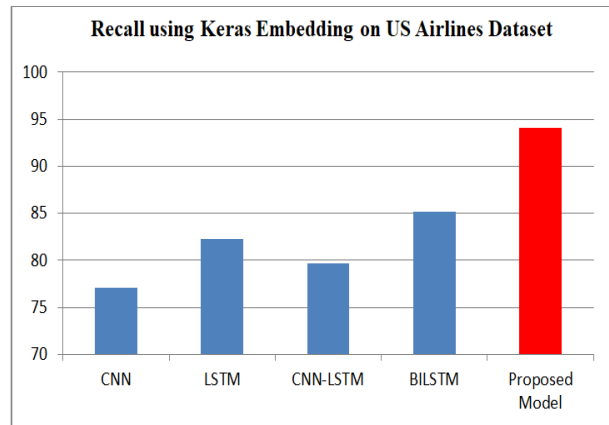
(c)



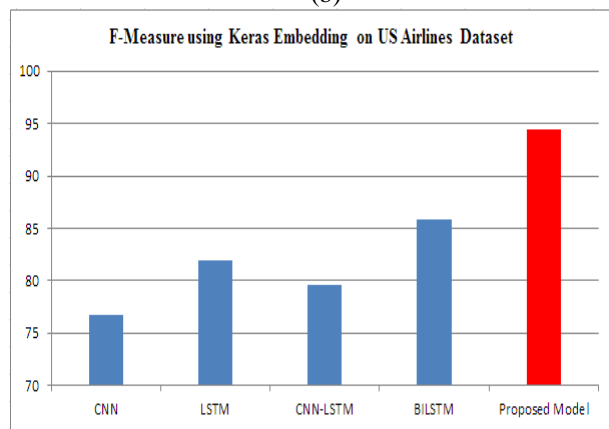
(d)

Figure 8: Model performance using Word2Vec word embedding for US Airlines dataset

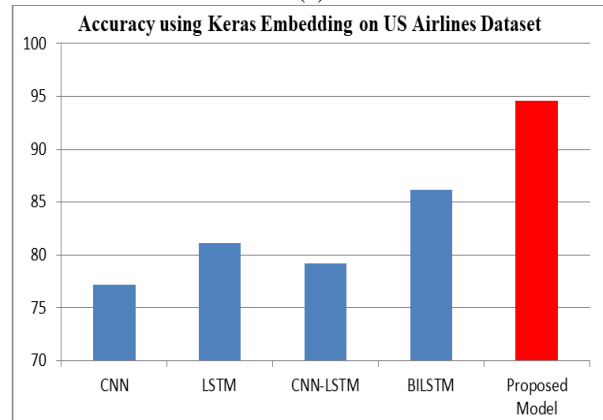
Since the proposed model performed much better on IR dataset using different word embeddings Word2Vec and Keras. Therefore, further experiments were conducted for testing the validity of the performance of the proposed model. The proposed model was then implemented on the US Airlines dataset using both word embeddings that were used for the IR dataset. The overall performance of the proposed model using Word2Vec embedding is shown in Figure 8 and Keras word embedding is represented in Figure 9. On the US Airlines dataset, the performance of the given self-attention-based model is observed to be reduced than the performance of the IR dataset but still, the proposed model outperforms other deep learning techniques in performance with 93.89%, 94.63% precision for Word2Vec and Keras Embeddings respectively. The classification accuracy on US Airlines dataset was also impressive and better than other models with 93.24% for Word2Vec and 94.56% for Keras Embedding.



(b)

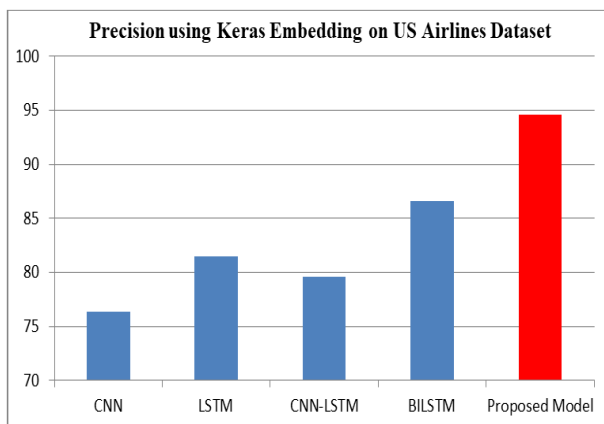


(c)



(d)

Figure 9: Model performance using Keras embedding for US Airlines dataset



(a)

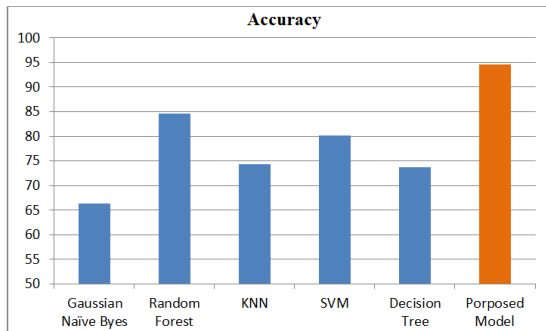


Figure 10: Accuracy of the proposed model over traditional models

Extensive experiments were performed for classic learning techniques on US airlines dataset for classification for validating the effectiveness of the model. Figure 10 demonstrates the accuracy level of the models. The proposed model outperformed the other traditional model also with a very high level of accuracy as compared to other models. The Gaussian Naïve Bayes shows the least performance among all with an accuracy of classification 66.60%. The Decision Tree method performed better than Gaussian Naïve Bayes with an accuracy of 73.5%. On the other hand, KNN, SVM, and Random Forest methods achieved an accuracy of 74.01%, 80%, and 84.5% respectively. The proposed model performed the most optimized accuracy with 94.56%.

5.3 Performance comparison with other state of the art model

The experimental findings were compared to earlier work in text sentiment classification methodologies to ensure that the performance model is verified.

Table 5: Experimental findings for sentiment classification accuracy in %

Models	Accuracy	Reported by
CNN-BiLSTM	90.66	Rhanou et al. [45]
CNN-BiLSTM	94.20	Zi-xian et al. [46]
BiLSTM with Self-Attention	86.24	Jun-Xie et al. [47]
BiLSTM with Muli-Head Attention	92.11	FEI et al. [48]
Conv-LSTM-Conv	89.02	Ghorbani et al. [49]
Text-CNN	91.50	Chuantao et al. [50]
CNN-BiLSTM with Keras	95.98	Proposed Model

Rhanou et al. [45] suggested a model that combines CNN with BiLSTM models using Doc2vec embedding for long text emotion analysis. The composite neural network model presented in [46] is comprised of two parts: a convolutional neural network for extracting local features from text vectors, a BiLSTM that extracts globalized features linked to context of text, and a fusion of the attributes collected by the two complementary models.

The sentences is automatically classified by the trained neural network based hybrid model. The results of experiments reveal that the accuracy rate of text classification is 94.2%, with a total of 10 iterations. For polarity classification of fine-grained sentiment in small sized texts, a BiLSTM model based on Self-Attention-Based using information of aspect-term is presented in Jun-Xie et al. [47]. A layer of word-encoding, a Dual LSTM module, an attention-based module, and a softmax function module are the primary constituents of the model. The vector based on hidden feautres and the vectors of aspects are merged by inserting in the BiLSTM module and the module based on self attention reducing computational complexity imposed by direct vector division. The model [47] achieved 86.24 accuracy which is lesser in comparison to proposed model. The model described in [48] investigates analysis of sentiment for Chinese text on social media by merging Multi-head Attention (MHAT) mechanism with BiLSTM networks for addressing the shortcomings of classic sentiment analysis. The goal of researchers was to add weights of influence to the generated sequence of text due to the MHAT mechanism's ability to learn important information from a distinct representation subspace utilising numerous dispersed computations. The model presented in [48] provided 92.11% accuracy. Gorbani et al. [49] suggested a ConvNet with BiLSTM model that classifies features using CNN, learns context information using BiLSTM, and then reuses the results for CNN to produce an abstract feature before applying to the final dense layer. The model achieved a great accuracy of 89.02 percent. The proposed model, on the other hand, is simpler and requires less complexity analysis, yet it achieves a greater accuracy of 6.96 percent more than [49]. Chuantao et al. [50] presented the BiLSTM deep learning model with two weak-tagging stages. The suggested approach employed weak-tagging for training the proposed model, lowering the detrimental influence of samples of noise in weak-tagging for the categorization of sentiment model's performance of categorization, and increases the accuracy of the sentiment categorization approach, which achieves 91.50 percent accuracy. In comparison to [50], the suggested Keras embeddings model outperformed the [50] model.

Based on the comparison with these previous models it is evident that the proposed model with Keras embeddings outperforms other models and achieves much better accuracy.

6 Conclusion

The research was performed on two different datasets. Word2Vec and Keras word embedding methods were applied for training and evaluation of the model on both datasets. The proposed model integrated the features of CNN with BiLSTM with the self-attention mechanism. ConvNet collects text characteristics and passes text context information to BiLSTM. The attention mechanism improved the classification accuracy as it extracted the context of the sentence more accurately. Hyper-parameters tuning was performed to optimize the model.

Therefore, the proposed model performed classification with improved accuracy and efficiency. The model provided improvement in accuracy by 7.20% using Word2Vec embedding for IR dataset and 9.84% more accuracy in classification using Keras embedding. The model performed effectively on US airlines dataset as well with 8.77% more accuracy with word2vec embedding and 8.03% more accuracy for Keras embedding. The proposed model outperformed other traditional models for the US Airlines dataset.

References

- [1] Liu B, *Sentiment Analysis and Opinion Mining* (Synthesis Lectures on Human Language Technologies), vol. 5, no. 1. San Rafael, CA, USA: Morgan & Claypool, 2012, pp. 1_167. Accessed: Nov. 1, 2020, doi: 10.2200/S00416ED1V01Y201204HLT016.
- [2] Tang HF, Tan SB and Cheng XQ, *Research on sentiment classification of Chinese reviews based on supervised machine learning techniques*. Chin. Inf. Process., vol. 21, no. 6, pp. 88_126, 2007.
- [3] Liu Y, J.W. Bi, and Z. P. Fan. *A method for multi-class sentiment classification based on an improved one- vs-one (OVO) strategy and the support vector machine (SVM) algorithm*. Inf. Sci, vols. 394_395, pp. 38_52, Jul. 2017.
- [4] Zhang J and Zong C, *Deep neural networks in machine translation: An overview*. IEEE Intell. Syst., vol. 30, no. 5, pp. 16_25, Sep./Oct. 2015.
- [5] Yin W, Schütze H, Xiang B, and Zhou B, *ABCNN: Attention-based convolutional neural network for modeling sentence pairs*. Trans. Assoc. Comput. Linguistics, vol. 4, pp. 259_272, Dec. 2016.
- [6] Ansari A, Maknojjia M, and Shaikh A, *Intelligent question answering system based on artificial neural network* in Proc. IEEE ICETECH, Coimbatore, India, Mar. 2016, pp. 758_763.
- [7] Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, and Kuksa P, *Natural language processing from scratch*. J. Mach. Learn. Res., vol. 12 pp. 2493–2537, Aug. 2011
- [8] Levy O, Goldberg Y, and Dagan I, *Improving distributional similarity with lessons learned from word embeddings*. Trans. Assoc. Comput. Linguistics, vol. 3, pp. 211–225, May 2015
- [9] Liu G and Guo J, *Bidirectional LSTM with attention mechanism and convolutional layer for text classification*. Neurocomputing, vol. 337, pp. 325_338, Apr. 2019.
- [10] Bahdanau D, Cho K, and Bengio Y, *Neural machine translation by jointly learning to align and translate*. 2014, arXiv: 1409.0473. [Online]. Available: <https://arxiv.org/abs/1409.0473>
- [11] Rush A M, Chopra S, and Weston J, *A neural attention model for abstractive sentence summarization*. 2015, arXiv: 1509.00685. [Online]. Available: <https://arxiv.org/abs/1509.00685>
- [12] Hermann K M, Kocisky T, Grefenstette E, Espeholt L, Kay W, Suleyman M, and Blunsom P, *Teaching machines to read and comprehend* in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 1684–1692.
- [13] Ranjan R., Daniel A.K. (2021), *Intelligent Sentiments Information Systems Using Fuzzy Logic*, Information and Communication Technology for Intelligent Systems. ICTIS 2020. Smart Innovation, Systems and Technologies, vol 195. Springer, Singapore. https://doi.org/10.1007/978-981-157078-0_55
- [14] Jiang L, Yu M, Zhou M, Liu X, and Zhao T, *Target-dependent Twitter sentiment classification* in Proc. ACL, 2011, pp. 151–160.
- [15] Perez-Rosas V, Banea C, and Mihalcea R, *Learning sentiment lexicons in Spanish*. LREC, vol. 12, p. 73, May 2012
- [16] Zhang M, Zhang Y, and Vo D T, *Gated neural networks for targeted sentiment analysis* in Proc. AAAI Conf. Artif. Intell., 2016, pp. 3087–3093
- [17] Tang D, Qin B, and Liu T, *Aspect level sentiment classification with deep memory network*. Proc. Conf. Empirical Methods Natural Lang. Process. 2016, pp. 214_224.
- [18] Tang D, Qin B, Feng X, and Liu T. *Effective LSTMs for target-dependent sentiment classification* in Proc. 26th Int. Conf. Comput.Linguistics, Tech. Papers (COLING), 2016, pp. 3298_3307.
- [19] Ren Y, Zhang Y, Zhang M, and Ji D, *Context-sensitive Twitter sentiment classification using neural network*. In Proc. AAAI, Phoenix, AZ, USA, Feb. 2016, pp. 215_221.
- [20] Rosenthal S, Farra N, and Nakov P, SemEval-2017 task 4: *Sentiment analysis in Twitter*. Proc. SemEval, Vancouver, BC, Canada, Aug. 2017, pp. 502_518.
- [21] Fan F, Feng Y, and Zhao D, *Multi-grained attention network for aspect-level sentiment classification*. In Proc. Conf. Empirical Methods Natural Lang. Process., 2018, pp. 3422_3433.
- [22] Zhang Q, Lu R, *A multi-attention network for aspect-level sentiment analysis*. Future Internet, vol. 11, no. 7, p. 157, Jul. 2019.
- [23] Xu Q, Zhu L, Dai T, and Yan C, *Aspect-based sentiment classification with multi-attention network*. Neurocomputing, vol. 388, pp. 135_143, May 2020.
- [24] Meng W, Wei Y, Liu P, Zhu Z, and Yin H, *Aspect based sentiment analysis with feature enhanced attention CNN-BiLSTM*. IEEE Access, vol. 7, pp. 167240–167249, 2019.
- [25] Park H J, Song M, and Shin K S, *Deep learning models and datasets for aspect term sentiment classification: Implementing holistic recurrent attention on target-dependent memories*. Knowl.-Based Syst. vol.187, Jan. 2020, Art. No. - 104825.

- [26] Lin Z, Feng M, Santos CN D, Yu M, Xiang B, Zhou B, Bengio Y, *A structured self-attentive sentence embedding*. arXiv preprint 2017, arXiv: 1703.03130.
- [27] Chen H, Sun M, Tu C, Lin Y, Liu Z, *Neural sentiment classification with user and product attention*. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 21 September 2016; pp. 1650–1659.
- [28] Fu, X, Yang, J, Li J, Fang M, Wang H, *Lexicon-enhanced LSTM with attention for general sentiment analysis*. IEEE Access 2018, 6, 71884–71891.
- [29] Dou, Z Y, *Capturing user and product Information for document level sentiment analysis with deep memory network*. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 3 March 2017; pp. 521–526.
- [30] Ranjan R, Daniel A K, *A Deep Learning Model for Extracting Consumer Sentiments using Recurrent Neural Network Techniques*, Int. Jour. Of Com. Sci and Net. Sec., Vol. 21 No. 8 pp. 238-246, 2021
- [31] Yang Z, Yang D, Dyer C, He X, Smola A., and Hovy E., *Hierarchical attention networks for document classification*, in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., 2016, pp. 1480–1489.
- [32] Chen H, Sun M, Tu C, Lin Y, and Liu Z, *Neural sentiment classification with user and product attention*, in Proc. Conf. Empirical Methods Natural Lang. Process., 2016, pp. 1650–1659.
- [33] Zhang Z, Zou Y, and Gan C, *Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression*, Neurocomputing, vol. 275, pp. 1407–1415, Jan. 2018.
- [34] Deng D, Jing L, Yu J, and Su S, *Sparse self-attention LSTM for Sentiment lexicon construction*, IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 27, no. 11, pp. 1777–1790, Nov. 2019.
- [35] Ahmad H., Asghar M. U., Asghar M. Khan Z., A. and Mosavi A. H., *A Hybrid Deep Learning Technique for Personality Trait Classification From Text*, in IEEE Access, vol. 9, pp. 146214-146232, 2021, doi: 10.1109/ACCESS.2021.3121791.
- [36] Cheng J., Sadiq M., Kalugina O. A., Nafees S. A. and Umer Q., *Convolutional Neural Network Based Approval Prediction of Enhancement Reports*, in IEEE Access, vol. 9, pp. 122412-122424, 2021, doi: 10.1109/ACCESS.2021.3108624.
- [37] Wang C., Yang X. and Ding L., *Deep Learning Sentiment Classification Based on Weak Tagging Information*, in IEEE Access, vol. 9, pp. 66509-66518, 2021, doi: 10.1109/ACCESS.2021.3077059.
- [38] F. Xu, Z. Pan, and R. Xia, *E-commerce product review sentiment classification based on a naïve Bayes continuous learning framework*, Inf. Process. Manage. vol. 57, no. 5, Sep. 2020, Art. no. 102221.
- [39] Preethi, G.; Krishna, P. V.; Obaidat, M. S.; Saritha, V.; Yenduri, S. ,2017: *Application of deep learning to sentiment analysis for recommender system on cloud*. International Conference on Computer, Information and Telecommunication Systems, pp. 93–97.
- [40] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, *Deep contextualized word representations*, J. Assoc. Comput. Linguistics, vol. 1, pp. 2227_2237, Mar. 2018.
- [41] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, Proc. IEEE, vol. 86, no. 11, pp. 2278_2324, Nov. 1998.
- [42] 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.
- [44] F. Y. Zhou, L. P. Jin, and J. Dong, *Review of convolutional neural network*, Chin. J. Comput., vol. 1, pp. 35_38, Jan. 2017.
- [45] M. Rhanou, M. Mikram, S. Yousfi and S. Barzali, *A CNN-BiLSTM Model for Document-Level Sentiment Analysis*, Mach. Learn. Knowl. Extr. 2019, 1, 832–847; doi:10.3390/make1030048
- [46] Zi-xian Liu, De-gan Zhang, Gu-zhao Luo, Ming Lian, Bing Liu., *A new method of emotional analysis based on CNN-BiLSTM hybrid neural network*, Cluster Computing, 2019, https://doi.org/10.1007/s10586-020-03055-9
- [47] JUN XIE, BO CHEN, XINGLONG GU, FENGMEI LIANG, AND XINYING XU, *Self-Attention-Based BiLSTM Model for Short Text Fine-Grained Sentiment Classification 56789,-(.volV)*, IEEE Access, 2019, doi 10.1109/ACCESS.2019.2957510
- [48] FEI LONG, KAI ZHOU, AND WEIHUA OU, *Sentiment Analysis of Text Based on Bidirectional LSTM With Multi-Head Attention, 2019, IEEE Access, doi 10.1109/ACCESS.2019.2942614*
- [49] M. Ghorbani, M. Bahaghighat, Q. Xin, and F. Özen, *ConvLSTMConv network: A deep learning approach for sentiment analysis in cloud computing*, J. Cloud Comput., vol. 9, no. 1, pp. 9_16, Dec. 2020, doi: 10.1186/s13677-020-00162-1.
- [50] Chuantao Wang , Xuexin Yang , and Linkai Ding, *Deep Learning Sentiment Classification Based on Weak Tagging Information*, 2021, IEEE Access, doi 10.1109/ACCESS.2021.3077059