A Deep Learning Model for Extracting Consumer Sentiments using Recurrent Neural Network Techniques

Roop Ranjan† and A K Daniel[2]††,

Madan Mohan Malaviya University of Technology, Gorakhpur, U.P., India

Abstract
The rapid rise of the Internet and social media has resulted in a large number of text-based reviews being placed on sites such as social media. In the age of social media, utilizing machine learning technologies to analyze the emotional context of comments aids in the understanding of QoS for any product or service. The classification and analysis of user reviews aids in the improvement of QoS. (Quality of Services). Machine Learning algorithms have evolved into a powerful tool for analyzing user sentiment. Unlike traditional categorization models, which are based on a set of rules. In sentiment categorization, Bidirectional Long Short-Term Memory (BiLSTM) has shown significant results, and Convolution Neural Network (CNN) has shown promising results. Using convolutions and pooling layers, CNN can successfully extract local information. BiLSTM uses dual LSTM orientations to increase the amount of background knowledge available to deep learning models. The suggested hybrid model combines the benefits of these two deep learning-based algorithms. The data source for analysis and classification was user reviews of Indian Railway Services on Twitter. The suggested hybrid model uses the Keras Embedding technique as an input source. The suggested model takes in data and generates lower-dimensional characteristics that result in a categorization result. The suggested hybrid model's performance was compared using Keras and Word2Vec, and the proposed model showed a significant improvement in response with an accuracy of 95.19 percent.

Key words:
Deep Learning, QoS, Word Embedding, Emotion Analysis, Text Analysis

1. Introduction

The world has changed dramatically due to current developments. People's lives have become more and more dependent on their ability to use the internet. Currently, social networking services provide a convenient method of communication that allows users to communicate with one another, discuss everyday activities, and elicit their opinions on several topics such as companies, goods, and locations on social forums. It also becomes an important source of information on people's opinions on a variety of topics. Customer reviews on social networking platforms might assist people in sharing inquiries and opinions about products prior to using them. Acquiring and analyzing data is beneficial to corporate growth, but only when it is analyzed and processed properly. The method of evaluating people's emotions and views from text using text analysis tools is known as sentiment analysis.

Machine learning [1], paraphrase classification [2], question answering [3], and sentence synthesis [4] are some of the natural language processing applications where neural networks are widely used. Neural networks have attained state-of-the-art performance in these tasks. However, when it comes to sentiment analysis, neural networks are still in their development. The widespread use of deep learning models in these days has aided the advancement of word processing and speech recognition. Deep learning models has been extensively employed in natural language processing [5], [6], including sentiment analysis, because of its automated learning properties. Machine learning based on Statistical techniques work efficiently in simple emotion classification applications but they are not suitable for analyzing difficult text analysis problems [7]. Deep learning models have shown to be significantly effective in sentiment classification [8], recognition of speech [9], and computerized vision [10]. Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) are two of the most extensively utilized deep learning algorithms in sentiment analysis.

Social media platforms play a great role for estimating quality of services of any product. Quality of Service (QoS) refers to a perspective of customer for expectations of service to a organization's execution. An organization well-known for its quality of service is more capable of handling user's expectations while being economically sound in its area. Various researches have performed analysis for quality of services for various industries. CNN and RNN is one of the emerging technique used to estimate quality of services. However, CNN is unable to learn the correlation sequence, and the performance of the CNN model is primarily determined by the window size used [12].RNN is a deep learning model that excels in constructing sequential models, but it is unable to extract local features in parallel. Since it can memorize the progression of flow of information over a time period, RNN becomes an alternative technique to CNN. Unfortunately, Recurrent Neural Network has a number of difficulties with gradient bursting and vanishing [12]. Because of these issues, RNN is difficult in training for long-sequence correlation for a given series. BiLSTM

Manuscript received August 5, 2021
Manuscript revised August 20, 2021
https://doi.org/10.22937/IJCSNS.2021.21.8.32

is a model based on RNN that has lately shown promising results in text based sentiment classification. It has dual LSTM orientations for helping the network better understanding of contexts accessible to it. BiLSTM has both forward and backward hidden layers, allowing the network for accessing sequence's preceding and subsequent contexts [13].Text sentiment classification, on the other hand, represents the text in the manner of vectors, which are often stored in a large-dimensional area. The moment BiLSTM retrieves background knowledge from aspects; it is unable to prioritize the most valuable data [14].CNN, unlike BiLSTM, contains a convolutional layer that extracts vector features and reduces their dimension. To address the aforementioned restriction, the goal of this research is to introduce a new text categorization deep learning based model that combines the structure of BiLSTM and CNN (CovNet and Dual-LSTM). The novel proposed CovNet-DualLSTM model tries for overcoming the limitations of Dual LSTM by including a convolution layer in the Convolution Neural Network model. The proposed method utilizes the efficiency and accuracy of Keras. The Keras API is based on a architecture that simplifies to construct deep learning based models using the APIs Theano or TensorFlow[15]. Keras has an integrated Embedding layer for textual data that may be utilized for recurrent neural networks. It's a robust model that can be utilized in a variety of ways, including learning a word embedding on its own, saving and restoring it in a later layer, and loading a word embedding model which is pre-trained , a sort of transfer learning. The model has been trained and on dataset used in Roop Ranjan et al.[16]. The dataset consists of 5000 tweets posted by travellers for services of Indian Railways. The proposed CovNet-DualLSTM model outperformed other models and earlier studies, according to the experimental observations.

The major contribution to this work is as below:

- In this research a novel hybrid mechanism is proposed for word embedding using Keras APIs.
- A model using CNN and DualLSTM is proposed to analysis and classify user's review.
- Rigorous experiments were performed on Proposed Hybrid model using Keras and Word2Vec embedding model and efficiency of the model has been compared with performance of existing models.
- The proposed model has shown a significant improvement in terms of accuracy and precision.

Further in Section II literature review of previous works is discussed. Section III presents the proposed model for sentiment analysis, while Section IV presents observation of experiments on the dataset and summarizes the findings. Section V and VI contains the conclusions, limitations, and recommendations for future research respectively.

2. Related Work

Text based Sentiment classification and natural language processing methods are extensively implemented for analyzing the emotional aspect of text. In their dataset, Yu et al. [17] created emotional resources using Chinese language for different categories of opinion labeling: arousal and behavior. As a result, they take into account of the behavior of positivity/negativity, also the arousal of highs/lows. Wu et al. [18] suggested a strategy classification of filtering important language of association, features and enhancing feature precision by reducing noise incrementally. Cui et al. [19] looked at the issue of n-gram order insensitivity. To increase sentiment analysis quality, their research exploited distributed semantic features of part-of-speech sequences. Fattach [20] investigated various term weighting schemes for sentiment analysis using a combination of many classifiers, with findings that surpassed their term weighting methods. Convolution Neural Network can learn large-dimensional and complicated classification using the multi-layer perceptron structure in deep learning.Xia et al. [21] extracted features from review texts using the conditional random field approach, and a SVM(support vector machine) was utilized for classification of sentiment for the reviews.

Deep learning models for sentiment analysis for textual data progressively became a pattern these days, thanks to the fast growth of deep learning methodologies .For example, Irsoy et al. [22] conducted sentiment analysis of news headlines using a Recurrent Neural Network (RNN), and their F-measure score achieved the range from 0.6 to 0.7. Recurrent Neural Network and LSTM-RNN cells were utilized by Liu et al. [23] to do sentiment classification of reviews of restaurants by consumer. Researchers discovered that RNN classification was marginally more effective than LSTM classification in this investigation. Restaurant remarks received an F1 score of 0.79 to 0.84, whereas notepad comments had an F1 value of 0.72 to 0.75. Tang et al. [24] introduced a target-based LSTM network, which uses BiLSTM networks for learning representation from the right and left context of an aspect. To model the syntax and semantic meaning of tweets, as well as the interplay between context and aspect, Zhang et al. [25] used neural network using gated topologies. Interaction based Attention Network) IAN learns the attention relationship among phrases and aspects target by an interactive technique, and it can retrieve features that are in relation with the aspect target using the attention mechanism.

As an important direction of investigation, researchers have integrated LSTM with different network structures. To overcome the difficult challenges in sentiment analysis, Kolawole [26] adopted the CNN-LSTM model. This method is simple to implement and can be performed without a lot of optimization in parameters. Wang et al. [27] developed a regional CNN-Long Short Term Memory architecture with two components: local CNN and LSTM, which can be applied for predicting the value of behaviour-arousal (BA) of texts ratings. The local CNN broke the input sentence into numerous regions and single sentence is used as an area to extract substantial sentimental information in each region, which was then weighted based on its contribution to the VA prediction. Wang et al. [28] proposed the ATAE-LSTM and AE-LSTM neural networks, which are both LSTM networks i.e. attention-based. The LSTM neural network used as Long Shot Term Memory based neural network for modeling the aspect, then paired the hidden state with the context embedding to construct the attention based feature vector, and then used a classifier for determining the sentiment category of the aspect. The ATAE-LSTM network, which is based on AE-LSTM, improved aspect embedding effects even more.

Several researchers have utilized deep learning to improve sentiment classification in recent years, with positive results. Socher et al. [29] introduced the RNTN (Recursive Neural Tensor Network) model, which included a sentiment tree library that synthesized meanings on the syntax tree of dual sentiment classification and produced an effective sentiment classification results in a movie review dataset. The CharSCNN [30] (Character to Sentence based Convolutional Neural Network) model extracted features of linked words and phrases using two convolutional layers and analyzed semantic knowledge for improving sentiment classification of small texts like tweets. The attention mechanism was incorporated into the LSTM by Baziotis et al. [31], which performed well in the sentiment classification of Task4 of SemEval-2017 for Twitter. Amir et al. [32] created a mode based on deep learning network that prioritizes context based data using lexicon based and grammar based patterns. The network that they developed learns user embedding which aids in enhancing sarcasm detection based on context. LeCun et al. [33] suggested using a CNN based model to retrieve text attributes and learn responses from spatial or temporal data. To minimize computation complexity and training parameters, he adopted the sharing of weight strategy in the CNN model.

3. Proposed Model

The proposed model consists of following sub models for the analysis.

1. Data Collection and Pre-Processing
2. Word Vectorization using Keras
3. CovNET-DualLSTM Model
4. Dense Layer Model
5. ReLU Activation Model

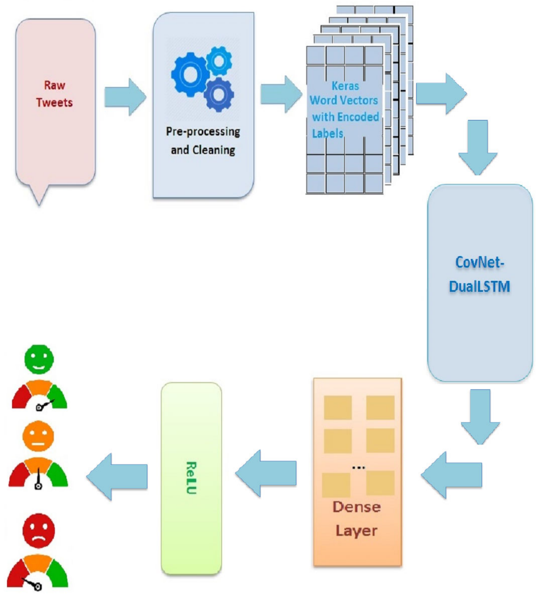

Fig. 1 Proposed Architecture.

3.1 Data Collection and Pre-Processing

The model used Twitter as the data source for retrieving the reviews. A registered developer account on twitter is used for collecting tweets using Python based Tweepy API. Tweets are the reviews of travellers travelling through Indian Railways. The data gathered from tweets contains noise, missing values, and is in an unstructured format therefore cannot be used directly to the model. So, it is required to pre-process the data before implementing to the model. Here python based library Pandas and Numpy has been used to pre-process the data by detecting and resolving missing values, categorical data encoding, splitting of the dataset.

3.2 Keras word Vectors with Encoded Labels

Word embedding is a technique for representing written words in forma of a set of numerical vector values. It generates same type of vector representations for texts with same type of meanings.. Each text is transformed to a single vector for providing input to a neural network. The API for word embedding is used here is Keras. Keras consists of a Word Embedding module for textual data that is utilized with neural networks. Integer-based encoding of the input data is required. Where a distinct number is used to represent each word. A TokenizerAPI is used provided with Keras. The dataset used here are in form of sentences. Following representation is used for performing Word Embedding using Keras:

Here every text consists of n words is which is represented by $V = \{x_1, x_2, \dots \dots \dots . . x_n\}$ and every word in the set is transformed into vector of words of dimension p. The input text is defined here as follows:

$$V = \{x_1, x_2, \dots \dots \dots . . x_n\} \in \quad (1)$$

Length of each document is not same always. Since the Keras Embedding layer requires all individual documents to be of same length. Therefore it is require making length of input text of same length (denoted as k).Hence inputs are padded for the shorter text with 0. Its length was extended by employing a zero-padding method. The text will be trimmed if it is longer than the predefined length l. Let us consider a text example of different documents of different length up to maximum length 12.

Table 1: Documents with different length

The encoding for document 1 is : [55, 15, 32, 29, 34, 33]
The encoding for document 2 is : [24, 2, 29, 34, 33, 37, 32, 29, 3]
The encoding for document 3 is : [22, 29, 16, 26, 32, 33, 7, 2, 2, 3, 33, 32]

The documents represented in Table 1 shows that their length is different from each other, therefore padding is required. Table 2 shows the structure of document after padding.

Table 2: Documents with same length after padding

The encoding for document 1 is : [55, 15, 32, 29, 34, 33, 0, 0, 0, 0, 0, 0]
The encoding for document 2 is : [24, 2, 29, 34, 33, 37, 3 2, 29, 3, 0, 0, 0]
The encoding for document 3 is : [22, 29, 16, 26, 32, 33, 7, 2, 2, 3, 33, 32]

The above implementation can be understood by the following figure 2.

55	15	32
29	34	33
33	0	0
0	0	0

(a)

24	2	29
34	33	37
32	29	3
0	0	0

(b)

22	29	16
26	32	33
7	2	2
3	33	32

(c)

Figure 2 : All documents in 4x3 uniform vector after padding, (a) Document 1 after 5 0's padding (b) Document 2 after 3 0's padding (c) Docuemtn3 , No padding required (Yellow shaded 0's representing padding)

In the above figure it is observed how all the documents have been uniformed to same size vector after padding process. Therefore dimension of all input sequence will be defined as follows:

$$V = \{x_1, x_2, \dots \dots \dots . . x_n\} \in S^{l*p} \quad (2)$$

3.3 Covnet-DualLSTM Model

This model combines the feature of Covent (CNN) and BiLSTM. The combinations of these are applied into the proposed model. The strength of the proposed hybrid model uses CNN convolution layers to extract the maximum amount of information from documents and provides it to the BiLSTM input, allowing the data to be kept in chronological order in forward as well as backward paths. This model can be explained through Covnet Model and DualLSTM Model as follows:

3.3.1 Covnet Model

This layer is used for extracting the most significant words from tweets. This model receives an embedded vector matrix

$$V \in S^{l*p} \quad (3)$$

The set V is provided as input given by Keras word Embedding Model. The model comprises a list of convolutional kernels with width μ and number of filters G that are convolved with the input text (N-dimensional metrics) to produce an output feature map.
Filter Gk, where $1 \leq k \leq K$ produces set of feature maps as shown below:

$$h_i^n = f(x^n \oplus Y_{i:i+x-1} + b^n) \quad (4)$$

For filter G_k the weight matrix is represented here as
$\alpha \in S^{e*p}$ and
b^n is associated bias of the filter G_k,
p is word vector dimension
$\oplus$ represents the operation of convolution.
The term $Y_{i:i+x-1}$ shows that the feature $Y_{i:i+x-1}$ from Y_i is extracted from Gk.
Nonlinear activation function represented by f
h_i^n is output for the feature map of filter G_k where each element i belongs to h^n.

Here ReLU (Rectified Liner Unit) function is applied to non-linear activation function f. Following feature maps were obtained for the sentence length l:

$$h = [h_1, h_2, \dots \dots \dots \dots \dots \dots \dots h_n] \quad (5)$$

The sliding process is continued till there are no more values to be covered further.

3.3.2 Pooling Model

Pooling Model is used for reducing the dimensions of the feature vectors. The feature vectors which are generated are optimized here using pooling layer. The convolution process output is used as input to the pooling process to execute for extracting the important features. There are various pooling techniques such as Max Pooling, Average Pooling and Global Pooling. The process of Max pooling is considered here for implementation because Max pooling provides the most optimized values for the given region of CovNet convolution.

$$\hat{h} = \max(h) \tag{6}$$

Here $\hat{h}$ is the set of optimized features extracted from feature vectors $h = [h_1, h_2, \ldots\ldots\ldots\ldots\ldots\ldots\ldots h_n]$ provided by Covnet Model.

3.3.3 DualLSTM Model

The DualLSTM model deals with optimized feature vectors received from the pooling model. The output of max pooling layer is concatenated to create the input to DualLSTM model. Each layer of DualLSTM contains two hidden states to allow the flow of information dual directions as forward to backward and vice versa. The dual scanning process leads input data from both the previous and the next stages. The forward LSTM state obtains previous information, whereas the reverse LSTM state obtains subsequent information. The process is iterated from first layer to the nth layer and the output R is received as follows: The process is represented in figure 3:

$$R = [R_{fwd,} R_{back}] \tag{7}$$

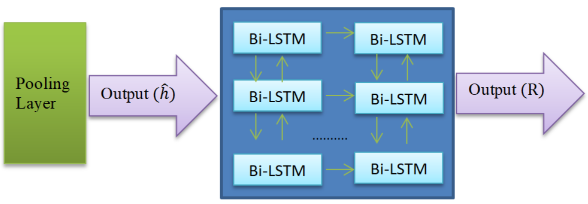

Figure 3: Process of DualLSTM Layer

3.4 Dense Model

The dense layer is used to minimize the error of the intermediate layers in backward direction using back propagation technique. A matrix-vector multiplication is performed and the errors in the values are trained and updated using back propagation. Updation at each layer is performed using outputs of all the previous layers.

3.5 ReLU Model

There are various activation functions utilized in neural networks like Sigmoid activation, Tanh activation and ReLU activation. An undesirable problem with Sigmoid and Tanh activation is when saturation of neuron's activation occurs at either end of 0 or 1, the values of gradient in these areas is virtually nil which leads to inaccurate calculations of feature vectors. Therefore the proposed model employs ReLU activation function t overcome this problem for generating the optimized feature vector map classification. The result of prediction by ReLU function is shown in Equation (8).

$$f(c) = \max(0, b) = \max\,(0, \textstyle\sum_{k=1}^{m} \alpha_i * c_i + e \tag{8}$$

This model receives the feature maps and produces Optimized classification results.

4. Experimental Results

This section discusses about the results achieved from the proposed model and performs a comparative analysis of the performance of the system with other proposed methods implemented in the same field.

Dataset is taken from Roop Ranjan et al.[16]. The dataset has been taken and pre-processed for 5000 tweets and categorized in positive, negative and neutral categories. The dataset is divided in three sets namely training dataset, testing dataset and validation dataset in proportion of 60:18:22 percent using Gaussian distribution.

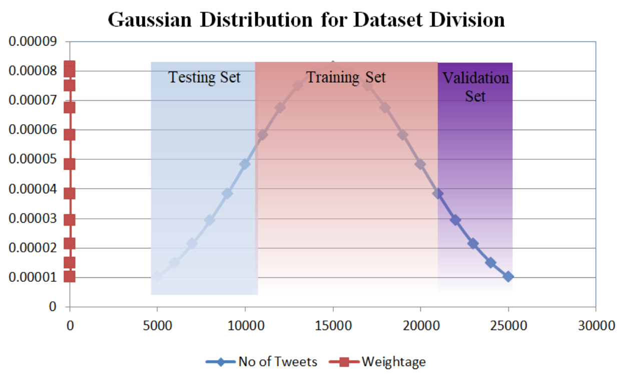

Figure 4: Dataset division using Gaussian Function

The dataset is categorized further in Positive, Negative and Neutral categories for Training, Testing and Validation set as shown in Table 3.

Table 3: Categorized Final Dataset

Date	*Positive*	*Negative*	*Neutral*
08-10-2019	2182	1098	1720
10-10-2019	2154	1074	1772
12-10-2019	2078	1052	1870
14-10-2019	2125	1048	1827
16-10-2019	2156	1080	1764

Table 4: Training Set

Date	*Positive*	*Negative*	*Neutral*
08-10-2019	1309	659	1032
10-10-2019	1292	644	1063
12-10-2019	1247	631	1122
14-10-2019	1275	629	1096
16-10-2019	1294	648	1058

Table 5: Validation Set

Date	*Positive*	*Negative*	*Neutral*
08-10-2019	393	198	310
10-10-2019	388	193	319
12-10-2019	374	189	337
14-10-2019	383	189	329
16-10-2019	388	194	318

Table 6: Testing Set

Date	*Positive*	*Negative*	*Neutral*
08-10-2019	480	242	378
10-10-2019	474	236	390
12-10-2019	457	231	411
14-10-2019	468	231	402
16-10-2019	474	238	388

4.1 Setting HyperParameter

During the CovNet Model, the model produced less overfitting or underfitting on values of output data. Therefore to avoid the problem of overfitting and underfitting and to obtain high performance of model hyper-parameter settings are used to optimize the performance. To improve accuracy, the randomized search approach was applied. Table 5 provides a brief of the hyperparameters:

Table 7: Setting of HyperParameters

Parameters	*Values*
Size of Kernel	5
Dimension(Embedding)	Keras(300)
Filter Size	64
DualLSTM Output Size	32
Regularization function	L2
Activation	ReLU
Weight Constraints	Kernel Constraints(max norm is 3)
Batch Size	64
No of Epoch	20
Batch Normalization	Yes
Learning Rate	0.01
Optimizer	Adam

4.2 Performance Measurement

Performance of proposed Covnet-DualLSTM model with Keras was observed and compared with Word2Vec i.e. CNN, LSTM, BiLSTM, CNN-LSTM Models. The proposed model outperformed the other models with an accuracy of 95.13%. The performance comparison is shown in Figure 5 as follows:

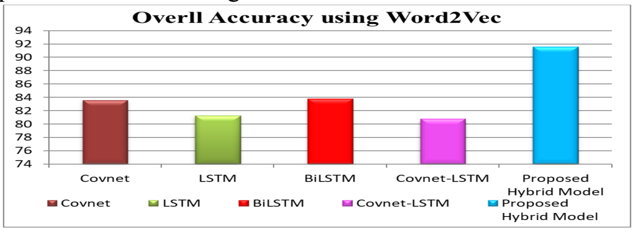

Figure 5: Level of accuracy using Word2Vec Embedding

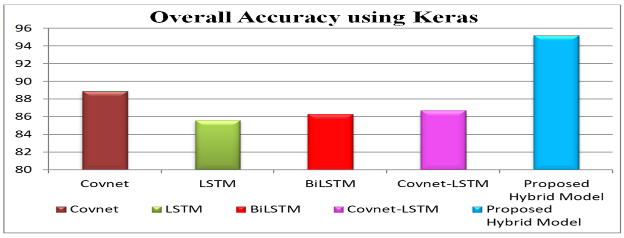

Figure 6: Level of accuracy using Keras Embedding

The above Figure 6 shows that for all the models that have been tested Keras embedding model proves to be a better performer than traditional Word2Vec embedding. It is also observed that when Keras embedding is combined with CovNet-DualLSTM, the accuracy level increases more and the accuracy level reaches to 95.19%. This clearly shows that the Keras embedding model with CovNet-DualLSTM out performs other existing models.

The performance parameters are validated using the Confusion Matrix with the help of SciKit-Learn API. Figure 7 illustrates parameters of a confusion matrix:

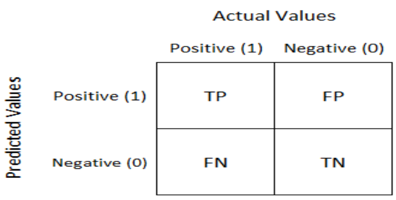

Figure 7: Parameters of Confusion Matrix

The performance and validation parameters are defined below:

True Positive (TP) - a result when the model predicts *positive* class is *correctly* [35]
True Negative (TN) - a result when the model predicts *negative* class is *correctly*
False Positive (FP) - a result when the model predicts *positive* class is *incorrectly*
False Negative (FN) - a result when the model predicts *negative* class is *incorrectly*
From the given Test Set following outcomes were retrieved as shown in Table 8.

Table 8: Predicted Values of Positive and Negative category

Total Tweets	***Total Tweets(Positive+Negative)***	***Total Positive Predicted***	***Total Negative Predicted***
7500	5894	3209	2685

The proposed model produced the following output using SciKit-Learn Matrix:

		Actual	
		Positive	Negative
Predicted	Positive	3055 (TP)	183 (FP)
	Negative	104 (FN)	2552 (TN)

Figure 8: Output of Confusion Matrix

Details of parameters of confusion matrix shows the improved performance of the Covnet-DualLSTM hybrid model over other models in Figure 7. The proposed hybrid model is tested with the review dataset and on the basis of the study with the optimum hyper-parameter tuning has been compared with other popular deep learning based models. The observation of comparison is presented in Figure 9.

Models	**Keras**				**Word2Vec**			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	89.33	87.65	91.56	91.65	89.65	90.78	90.78	89.95
LSTM	90.16	88.35	89.21	92.63	90.14	89.54	88.63	86.65
Bi-LSTM	87.32	91.65	91.13	90.65	86.65	92.63	89.51	88.37
CNN-LSTM	89.12	90.32	90.65	91.09	88.89	91.62	90.62	86.36
Proposed CovNetBiLSTM	**94.34**	**96.7**	**95.5**	**95.19**	**91.62**	**93.89**	**93.36**	**91.6**

Figure 9: Output of Confusion Matrix

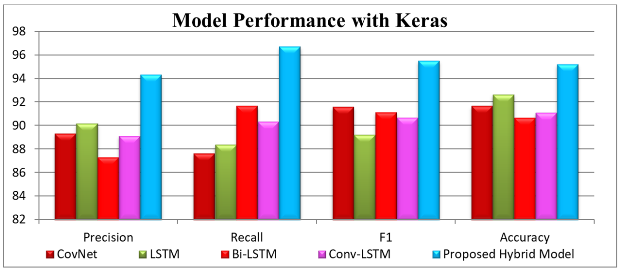

Figure 10: Graphical representation of Model Performance with Keras

The experimental results of performance of the proposed hybrid model with Keras Word Embedding Technique are shown in Figure 10. The hybrid model classification provides a higher level of accuracy and better results than other models. Keras effectively initialises word vectors for datasets, as indicated by the overall correctness of experimental findings. The accuracy level of 95.19% proves that the Keras word vectoring model affected the overall accuracy of proposed model.

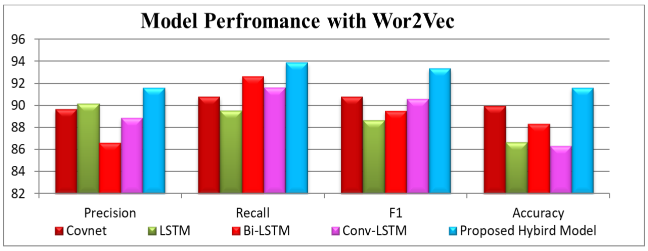

Figure 11: Graphical representation of Model Performance with Word2Vec

Figure 11 shows the performance of the system with Word2Vec Embedding. The above figure clarifies that hybrid model produces much better results as compared with CNN,LST, BiLSTM and Conv-LSTM.

Observation in Figure 10 shows that the proposed model used with Keras Embedding has outperformed other models with an improvement in accuracy of 3.54%, 4.18% in precision, 5.05% in recall and 3.94% over all the other models represented in Table 9.

Table 9: Performance Improvement using Keras

Parameter	**Performance Improvement**
Accuracy	3.54%
Precision	4.18%
Recall	5.05%
F1(F-Measure)	3.94%

Outcomes in Figure 11 show the improvement in performance of the system using Word2Vec embedding as compared to other models. Proposed model with Word2Vec Embedding shows improvement by 1.48% in precision, 1.26% in recall, 2.58% in F1 and 1.65% in accuracy represented in Table 10.

Table 10: Performance Improvement using Word2Vec

Parameter	**Performance Improvement**
Accuracy	1.65%
Precision	1.48%
Recall	1.26%
F1(F-Measure)	2.58%

It was also observed that is CNN and BiLSTM is used separately on different models, the performance does not provide effective results.

5. Conclusion

The dataset of reviews for Indian Railways is used to train and evaluate the model. The proposed model improves the system's efficiency and accuracy. On the suggested hybrid model, both Keras and Word2Vec embedding were utilised for word vectorisation separately, and the system's performance improved significantly. With a 3.54 % improvement in accuracy using Keras and 1.65 % improvement using Word2Vec, the model was able to accurately classify text sentiment. The findings of experiments validated the model's viability and effectiveness. In comparison to existing models, the proposed model performs significantly better.

6. Limitation and Future Work

The research was performed on the dataset of 5000 cleaned tweets. The volume of data can be increased to check the performance of the proposed model. The structure of the proposed Covent-DualLSTM model could be changed in the future to improve sentiment classification performance. Other techniques for pre-processing like POS tagging can be used to check for improvement in overall accuracy. The results of the experiments showed that word representation have an impact on the correctness of the entire model. As a result, combining features of different word embedding techniques may result in greater extraction of features for the network.

References

[1] J. Zhang and C. Zong, ‘‘Deep neural networks in machine translation: An overview,’’ IEEE Intell. Syst., vol. 30, no. 5, pp. 16–25, Sep./Oct. 2015.

[2] W. Yin, H. Schütze, B. Xiang, and B. Zhou, ‘‘ABCNN: Attention-based convolutional neural network for modeling sentence pairs,’’ Trans. Assoc. Comput. Linguistics, vol. 4, pp. 259–272, Dec. 2016.

[3] Ansari, M. Maknojia, and A. Shaikh, ‘‘Intelligent question answering system based on artificial neural network,’’ in Proc. IEEE ICETECH, Coimbatore, India, Mar. 2016, pp. 758–763.

[4] M. Rush, S. Chopra, and J. Weston, ‘‘A neural attention model for abstractive sentence summarization,’’ 2015, arXiv:1509.00685. [Online]. Available: https://arxiv.org/abs/1509.00685

[5] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, ``Natural language processing (almost) from scratch," J. Mach. Learn. Res., vol. 12 pp. 2493_2537, Aug. 2011.

[6] O. Levy, Y. Goldberg, and I. Dagan, ``Improving distributional similarity with lessons learned from word embeddings," Trans. Assoc. Comput.Linguistics, vol. 3, pp. 211_225, May 2015.

[7] Q. Huang, R. Chen, X. Zheng, and Z. Dong, ``Deep sentiment representation based on CNN and LSTM," in Proc. Int. Conf. Green Informat., Aug. 2017, pp. 30_33.

[8] J. Schmidhuber, ``Deep learning in neural networks: An overview," Neural Netw., vol. 61, pp. 85_117, Jan. 2015.

[9] L. Brocki and K. Marasek, ``Deep belief neural networks and bidirectional long-short term memory hybrid for speech recognition," Arch. Acoust., vol. 40, no. 2, pp. 191_195, Jun. 2015.

[10] V. Campos, B. Jou, and X. Giró-i-Nieto, ``From pixels to sentiment: Fine tuning CNNs for visual sentiment prediction," Image Vis. Comput., vol. 65, pp. 15_22, Sep. 2017

[11] H.-F. Tang, S.-B. Tan, and X.-Q. Cheng, ``Research on sentiment classication of chinese reviews based on supervised machine learning techniques,"J. Chin. Inf. Process., vol. 21, no. 6, pp. 88_126, 2007.

[12] 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.

[13] G. Liu and J. Guo, ``Bidirectional LSTM with attention mechanism and convolutional layer for text classi_cation," Neurocomputing, vol. 337, pp. 325_338, Apr. 2019.

[14] F. Y. Zhou, L. P. Jin, and J. Dong, ``Review of convolutional neural network," Chin. J. Comput., vol. 1, pp. 35_38, Jan. 2017.

[15] Official website of Keras," https://keras.io/about/"

[16] 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/10.1007/978-981-15-7078-0_55

[17] L.-C.Yu, L.-H. Lee, S. Hao, J.Wang,Y. He, J. Hu, K. R. Lai, and X. Zhang, ``Building Chinese affective resources in valence-arousal dimensions," in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang.Technol. (NAACL-HLT), 2016, pp. 540-545.

[18] J.-L. Wu, L.-C. Yu, and P.-C. Chang, ``Emotion classi_cation by removal of the overlap from incremental association language features," J. Chin. Inst. Engineers, vol. 34, no. 7, pp. 947_955, Oct. 2011.

[19] Z. Cui, X. Shi, and Y. Chen, ``Sentiment analysis via integrating distributed representations of variable-length word sequence," Neurocomputing, vol. 187, pp. 126_132, Apr. 2016.

[20] M. Abdel Fattah, ``New term weighting schemes with combination of multiple classifiers for sentiment analysis," Neurocomputing, vol. 167, pp. 434_442, Nov. 2015.

[21] H. Xia, Y. Yang, X. Pan, Z. Zhang, and W. An, ``Sentiment analysis for online reviews using conditional random fields and support vector machines," Electron. Commerce Res., pp. 1_18, May 2019, doi:10.1007/s10660-019-09354-7.

[22] O. Irsoy and C. Cardie, ‘‘Opinion mining with deep recurrent neural networks,’’ presented at the Conference Empirical Methods Natural Language Processing (EMNLP), 2014.

[23] P. Liu, S. Joty, and H. Meng, ''Fine-grained opinion mining with recurrent neural networks and word embeddings,'' presented at the Conference Empirical Methods Natural Language Processing, 2015

[24] D. Tang, B. Qin, X. Feng, and T. Liu, ''Effective LSTMs for target dependent sentiment classification,'' in Proc. 26th Int. Conf. Comput. Linguistics (COLING), Osaka, Japan, 2016, pp. 3298–3307.

[25] M. Zhang, Y. Zhang, and D. Tang, ''Gated neural networks for targeted sentiment analysis,'' in Proc. 30th AAAI Conf. Artif. Intell., Phoenix, AZ, USA, 2016, pp. 3087–3093.

[26] S. K. Sønderby, C. K. Sønderby, H. Nielsen, and O. Winther, ''Convolutional LSTM networks for subcellular localization of proteins,'' in Proc. 2nd Int. Conf. Algorithms Comput. Biol. Mexico City, Mexico: Springer-Verlag, 2015, pp. 68–80.

[27] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, ''Dimensional sentiment analysis using a regional CNN-LSTM model,'' in Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, Berlin, Germany, 2016, pp. 225–230.

[28] Y. Wang, M. Huang, X. Zhu, and L. Zhao, ''Attention-based LSTM for aspect-level sentiment classification,'' in Proc. EMNLP, 2016, pp. 606–615

[29] R. Socher, ''Recursive deep models for semantic compositionality over a sentiment treebank,'' in Proc. Conf. Empirical Methods Natural Lang. Process., Seattle, WA, USA, 2013, pp. 1631–1642.

[30] D. Santos and M. Gattit, ''Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts,'' in Proc. 25th Int. Conf. Comput. Linguistics: Tech. Papers, Dublin, Ireland, 2014, pp. 69–78.

[31] Baziotis, N. Pelekis, and C. Doulkeridis, ''Datastories at SemEval2017 Task 4: Deep LSTM with attention for message-level and topicbased sentiment analysis,'' in Proc. 11th Int. Workshop Semantic Eval. (SemEval), Vancouver, BC, Canada, Aug. 2017, pp. 747–754.

[32] S. Amir, C. B. Wallace, H. Lyu, P. Carvalho, and J. M. Silva, ''Modelling context with user embeddings for sarcasm detection in social media,'' in Proc. CONLL 2016.

[33] 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.

[34] Roop Ranjan, AK Daniel, Emotion Detection for Travelling Services using Rule-Based Fuzzy Inference System, Springer International Conference on Machine Intelligence and Smart Systems (MISS2020), Rustamji Institute of Technology, Tekanpur, Gwalior, M.P., September 24-25, 2020.

[35] Krishnani, A. Kumari, A. Dewangan, A. Singh and N. S. Naik, "Prediction of Coronary Heart Disease using Supervised Machine Learning Algorithms," TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), 2019, pp. 367-372, doi: 10.1109/TENCON.2019.8929434.