DistilBERT-CNN-LSTM Model with GloVe for Sentiment Analysis on Football Specific Tweets

Venkatesh, *Member, IAENG,* Siddhanth U Hegde, Satish B Basapur, and Nagaraju Y

***Abstract*— The football fan's feelings get unfold during the different phases of a football match; they express their opinions, views, thoughts, judgments of the game, attitudes towards players, and emotions, stances on social media like Twitter. The change in the fan's opinion is reflected in fans' tweets. This research focuses on identity and analyzing sentiment in tweets expressed by football fans on Twitter. We analyzed different word vector that captures semantic and syntactic information to perform sentiment analysis. This paper uses the global vector (GloVe) word embedding technique to produce word vectors with substructure and leverage statistics. In addition to GloVe, we also construct a sentiment lexicon as additional information. The Word vector produced by GloVe and sentiment lexicon are inputs to DistilBERT. DistillBERT is a language representation model that leverages distillation during the pre-training phase. The proposed DistilBERT-CNN-LSTM deep learning model blends the benefits of CNN and LSTM. The CNN is used to excerpt features from word embedding that reflect short-term sentiment dependency, while LSTM builds long-term sentiment relationships among words. This paper also used machine learning algorithms such as Random Forest, Support Vector Machine, Multinomial Nave Bayes, K-Nearest Neighbours (KNN), and XG Boost for sentiment analysis and sentiment classification. We evaluated the performance of the proposed DistilBERT-CNN-LSTM with the GloVe word embedding approach with the 2018 FIFA world cup tweets dataset; our experiment results show 86.5% and 92.56% validation and testing accuracy, respectively. Further, our experimental results demonstrate that the Random Forest algorithm performs more consistently and robust performance than other machine learning classifiers; it perceives fans' sentiments during football events.**

***Index Terms*— Global Vector, Feature Extraction, LSTM., Lexicon, Stance Analysis, Twitter.**

I. Introduction

With the swift advancement of e-commerce platforms and the internet, customers more often use e-commerce to buy the commodity or get services. Online shopping provides an option for customers to choose a commodity from a wide range of collections and styles. The consumers procure their craving products. Still, it also annoys the customer because of the discrepancy between the explanatory of the product and the actual product, unacceptable quality of the commodity, flaws in products after the transaction, and delivery of the commodity.

The customer uses online social networks (OSN) such as Twitter to share, express, and exchange their online shopping experience with others. Tweets on Twitter convey the views of users about political, social events, sports events, and products. It is essential to perform sentiment analysis on the views or opinions expressed by customers who have purchased products or got services and done a rigorous evaluation of the commodity or products or service he/she got on online or electronic commerce platforms. The stance or opinion detection methods are used extensively in the business and stock market to interpret customer opinion towards a product, stock, service, and market trend.

The sentiment analysis is the most common tool to determine the underlying sentiment present in the message, text, or feedback. The sentiment may be expressed by a layman or domain expert. The sentiment is generally written in natural language in structure or unstructured format; it may be semantically and syntactically incorrect. Still, customer sentiment, views, and opinions are hidden in it. The stance may be in the form of the voice of the customer, reviews, survey responses, online submissions, and tweets on Twitter.

The sentiment analysis can be applied in reviewing the emotions expressed by the fans of games, entertainment, amusement reality shows, and sports events such as the FIFA football world cup. The fans of soccer express their views or verbalize. The tweets on Twitter generate an enormous amount of data. The utility of online social networks such as Micro-blog, Facebook, LinkedIn, and Twitter has become a popular platform to express, discuss, and exchange opinions about the sports events that are going on. Football events such as the FIFA World cup attract millions of fans around the globe. They express their emotions at different phases of the event: goal-scoring time when players get injuries during play, the star player of the game, shots played by the player, chances of winning the game by a particular team, the outcome of the soccer match. They use social media as a representative tool to express their emotions through tweets, posts, etc. The sentiment analysis summarizes the intensified emotions exhibited by the fans for the soccer match, win or loss. It also gives us new insights into how fans' emotions change at different stages of the sport. The insights about a certain sports event

Manuscript received 20 October, 2020; revised 05 May, 2022.

Venkatesh is Assistant Professor of the Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bengaluru-560001, INDIA (corresponding author: phone: 080-22961836; e-mail: venkateshm.uvce@bub.ernet.in).

Siddhanth U Hegde is B. Tech student of Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bengaluru-560001, INDIA (e-mail: siddhanthhegde227@gmail.com).

Satish B Basapur is Assistant Professor, Department of Information Science and Engineering, Dr. Ambedkar Institute of Technology, Bengaluru-560056, INDIA (e-mail: satish.basapur@gmail.com).

Nagaraju Y is PhD Scholar of the Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bengaluru - 560001, INDIA (e-mail: nagarajuyat22@gmail.com).

help warn the authorities about the possibility of the violence due to its outcome or help analyze a particular player or haphazardly or a particular area team. This paper aims to a built hybrid model to analyze sentiment and analyze the performance of various ML classifiers in identifying hidden sentiments in football-specific tweets of the 2018 FIFA world cup.

Motivation:

- The conventional text document is syntactically written using formal language and formatted with a predefined structure. However, the text written on OSN Twitter is unstructured, syntactically, and semantically incorrect; tweets include slang words, abbreviations and a limited number of words on tweets.
- The standard sentiment classifiers fail to identify sentiment present in tweets due to unstructured, ambiguous tweets and informal language to express their views or emotions.
- The significance of a word, similarity, and co-occurrence of words is computed using the distance or angle between a pair of words that do not capture sentiment information.
- It is hard to accurately detect irony and sarcasm in conversations as people use positive texts to express their negative emotions.

Contributions:

- Prepared our dataset consisting of football-specific tweets of the 2018 FIFA world cup; all tweets in the dataset are pre-processed and labeled manually used extensively to perform analysis of stance. We collected 530k tweets specific to football events and 10007 tweets specific to FIFA World Cup 2018.
- Introduced the word vector representation method that gives co-occurrence statics and linear substructure prevalent in the corpus.
- Introduced a light-faster-small language representation model to identify vocabulary using context.
- We designed a hybrid CNN-deep learning model for the football-specific dataset to find hidden sentiments and accurately classify the football-specific tweets into different categories.
- ML classifiers' performance is analyzed to extract hidden sentiment present in football-specific tweets; a comprehensive experiment is conducted to investigate classifiers on different performance parameters.

Organization of Paper: This paper is formulated in the subsequent format. Section 1 gives an introduction. The detailed literature survey is presented in section 2. Section 3 explains the problem statement. The discussion on the proposed solution is given in section 4. Section 5 elaborates on the experiment results and analyzes the performance of the experiment results. Section 6 presents conclusions.

II. Literature Survey

The meticulous survey of current state-of-art research work from the prevalent literature gives more insights into sentiment analysis and classification of sentiments. A comprehensive study of set methods to identify sentiments in football-specific is carried out in this section.

The sentiment assessment is widely used in profuse domains such as recommendation systems, decision, and opinion support systems. Accurate sentiment analysis plays a vital role in the success of the e-commerce business, forming government policies, identifying trends in the stock market, etc. Sentiment analysis is an area of interest for researchers. Text is represented using a word embedding; Word embedding represent text in term of semantics rather than sentiment. For accurate sentiment analysis, text with sentiment information is useful. The accuracy of sentiment analysis is improved by using sentiment lexicons and word embedding, the position of the word, and phrases that comprise fuzzy words or sentiments [1-2].

To identify polarity in tweets of the 2014 FIFA World Cup. Barnaghi et al. [3] acquired football-specific tweets and applied a well-known text classifier to find the stance or opinion hidden in tweets specific to football events. The unique words of tweets are represented using bigram and unigram. The well-known classifier can categorize tweets' polarity with 72% accuracy. The word vector representation with unigram and bigram fails to capture statics of the corpus; it considers local context windows to represent the co-occurrence of words.

Similarly, Peiman et al. [4] determine a sentiment present in football-specific tweets; it uses TF-IDF to represent the importance of words or phrases in a document. It eliminates undesirable features such as noise from the dataset. The proposed model also established the relation between events and sentiments. However, TF-IDF does not reflect the position of words or phrases, and it depends heavily on the corpus. The proposed method in [3] and [4] has used the 2014 FIFA World cup dataset. In both methods, well-known text classification methods are used to determine the polarity present in tweets related to a football event. It also correlates tweets posted on Twitter and sports events held across the globe.

The challenges of performing sentiment assessment on big data and all-inclusive reviews on sentiment assessment techniques are discussed [11]. Authors in [5] blended the benefits of different deep learning models and word embedding techniques to identify and classify hidden sentiment in Arabic, Turkish, and Lithuanian texts. Similarly, authors in [6] combined the benefits of TF-IDF and RNN, LSTM, BiLSTM model to capture and analyze sentiments in text. Elbagir et al. [7] has used ordinal regression to perform sentiment analysis on Twitter texts. However, the word typifying methods in the state-of-art research on sentiment assessment do not examine the words' sentiment information. The Capsule network, Recurrent, Recursive neural network, and feature ensemble models are adopted for sentiment assessment [8-9-10].

The Word embeddings techniques haul semantic information from words instead of sentiment information. Traditionally, the structure of a sentence is represented with a bag-of-words; the bag-of-words are used to train a classifier and assess sentiments. However, bag-of-words discard semantic information and the ordering of words. N-gram is another traditional approach; N-gram orders words and represents words in high-dimensional space. However, the score assigned to words are distinct and suffer from sparsity.

Fatimah et al. [12] have presented a stance detection method that considers a non-domain specific and unbalanced dataset. Twitter's tweets contain hashtags, abbreviations, slang words, and unstructured and informal language. Therefore, it is essential to label all datasets manually. All 30 million tweets related to the 2014 FIFA World cup are labeled. The SVM classifier can classify Twitter text with better accuracy for the automatically labeled dataset. The author has analyzed the performance of SVM with the manually labeled data set and automatically marked data set using API.

Due to its simplicity and classification, the accuracy k-NN classifier is widely accepted. However, the traditional k-NN classifier cannot handle a large dataset and huge features. Overcoming these limitations, the author in [13] has modified the k-NN classifier to handle a large sample size of medical data and perform classification with better accuracy. The improved k-NN performs clustered-based pre-processing and reduces samples by clipping data into different categories. Jonathan et al. [14] proposed a classifier for 2014 FIFA world cup tweets. The lexicon is constructed based on tweets' semantic orientation, and a piece of tweets is represented as a bag of words. The proposed method identifies sentiment among a large set of words and also indicates a multitude of statistics. The author analyzed the Naive Bayes classifier's performance on the dataset of SemEval 2014 [15]. Aloufi et al. [16] have considered tweets related to English Premier League Soccer 2016 and UEFA Champions League Soccer 2016/17. The author used Bag of Words (BOW), lexicon, and linguistic features to represent the word vector and trained model. The author has not achieved better accuracy of classification when dictionary, Bag-of-Words (BOG), and language-specific features are used to represent word vectors.

Wang et al. [17] have explained the different phases of the sentiment analyses model, various sentimental models, their complexity, the general framework, and the evaluation parameters for the model. The author demonstrated that a method that combines the POS tagger, syntactic, and semantic tree with the classifier algorithm would result in better classification. The author also highlighted that non-availability of the lexicon, unstructured language, and non-domain-specific tweets impede sentiment analysis performance. Wijayanto et al. [18] used the Gaussian Nave Bayes approach to categorize customer reviews on product quality, the taste of food, and the type of movies. The value is categorical; the author has used the Chi-Squared test for feature selection, stop-words, and removed commonly used words.

Aloufi et al. [19] have considered various classifiers to categorize and analyze football fans' stances; the dataset is a collection of tweets related to football events, tweets specific to the 2016/17 Champions League and 2014 FIFA World Cup. The author has created a football-specific lexicon containing 3,479 unique words and labeled it according to fan stance. The classifier has used 54,526 manually labeled tweets to determine the stance of football fans. The experiment results illustrate that SVM exhibited robust and consistent performance compared to RF and MNB classifiers.

Nurmaini et al. [20] have proposed a Faster CNN model to determine lesions in CT images. Authors in [21] have analyzed the importance of activation functions used in CNN. The author also summarized the performance and selection of activation functions in GooglNet, VGGNet, Network in Network(NIN), and AlexNet.

Caiwen et al.[22] used sparsity of neighborhood and weighting scheme to classify the imbalanced data.

III. PROBLEM STATEMENT

The existing distinct word embedding techniques are not efficient, accurate enough to leverage the sentiment information count of co-occurrence of words in tweets specific to the 2018 FIFA world cup. Design and use a word embedding technique that builds word vectors on co-occurrence and sentiment information. The word vectors are input to the hybrid CNNdeep learning model. Develop a hybrid model that integrates the benefits of CNN and the deep learning model. The hybrid model must handle short tweet text sequences and long tweets text sequences and automatically analyze the emotion of football fans' sentiments hidden in football-specific tweets.

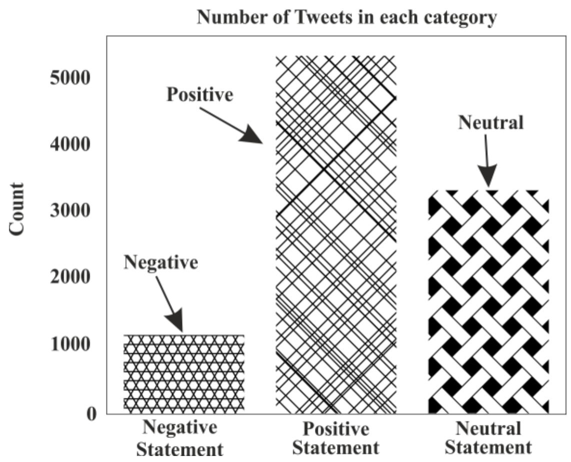


Fig. 1. Number of Tweets considered during conduction of experiments

IV. PROPOSED METHODOLOGY

A. *Materials and Methods*

Tweets related to the 2018 FIFA Worldcup are gathered using Twitter API to conduct experiments. Our dataset includes N tweets; it is represented as Dataset = T_1 , T_2 , T_3 , T_N, where n is 10,007 tweets considered for the test and training dataset. Each tweet is manually annotated and attributed a sentiment to each tweet. For every tweet following sentiment class is assigned manually:

Positive sentiments: "Goal is priceless", "Amazing player", "what a shot", "spectacular goalkeeping", "Best team in the world right now". "The man for the big occasion", "a free-kick", "scores the winning penalty to send them into the quarter-finals where they will face Russia". Negative sentiments: "cowardly coach", "Boredom", "Hate player who made several mistakes", "Sadness of game", "worry of the strong opponent team", "only 1 penalty has been saved by the goalkeeper". Neutral sentiment: "Register with Coral via the link below to register for Morocco".

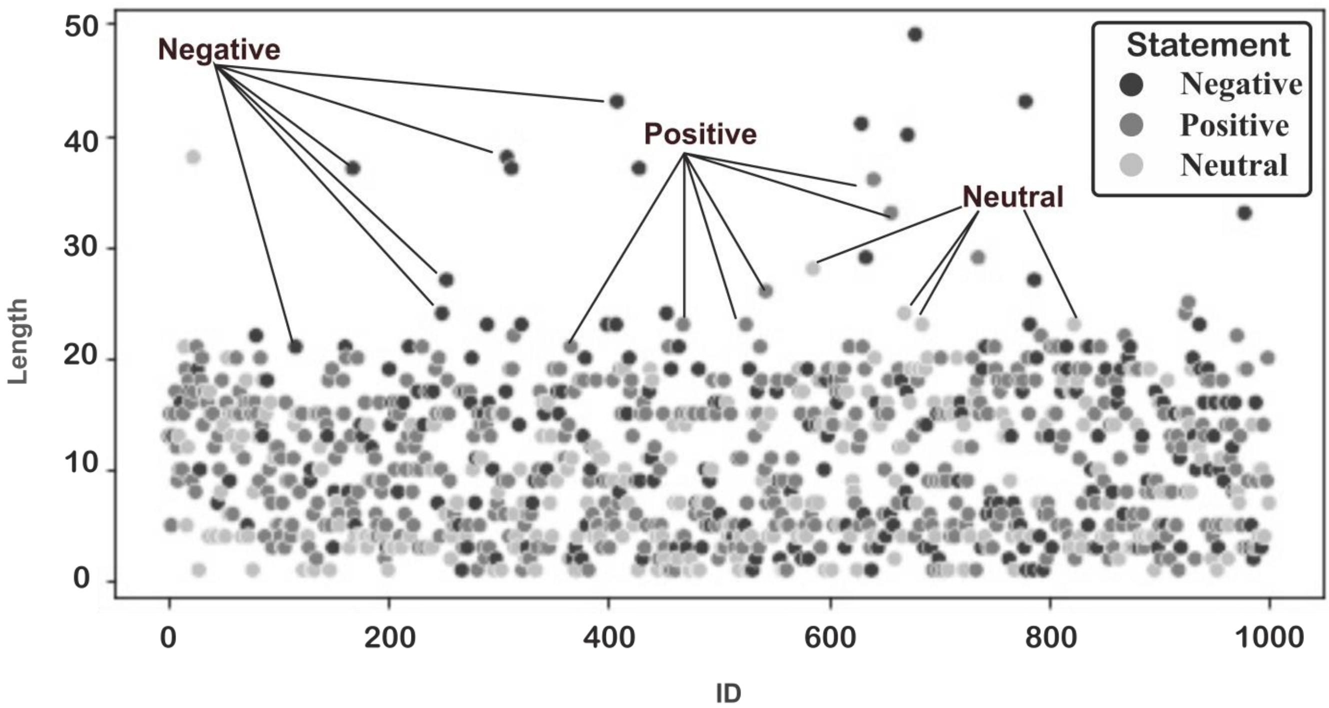


Fig. 2. Tweets Distribution Order

The dataset is fission into a training, test, and validation test. The training dataset is engaged to train the proposed model. A test set is used to execute experiments. The classification report was obtained from this dataset. The validation test is conducted to validate performance parameters such as accuracy, precision, sensitivity, and specificity. Figure 1 shows the number of tweets considered for our experiments; the number of positive tweets is more than the number of negative tweets. Figure 2 depicts distribution of tweets in the dataset, it is observed that the average length of a tweet is about 15 to 16 and few tweets have a length of 40 to 50.

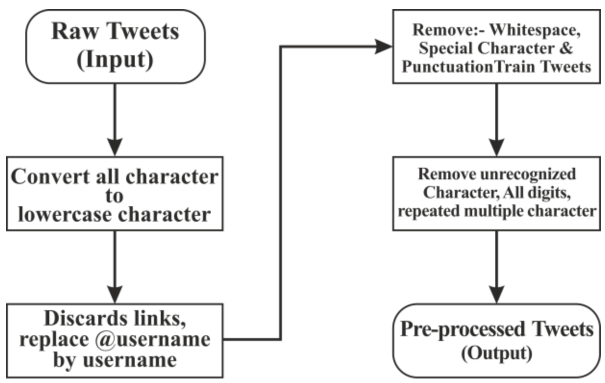


Fig. 3. Pre-processing Tweets

B. *Data Pre-processing*

The tweets posted by users or football fans on Twitter contain hashtags, abbreviations, slang words, URLs, and unstructured and incomplete statements. These are noise in tweets posted on Twitter. The incomplete, unstructured tweets, compressed and noise in tweets are pre-processed using steps shown in figure 3. All uppercase characters were converted to lowercase, discarded link in the tweets, convert @username to username, eliminated multiple spaces, punctuation, number, unrecognized characters, correct spelling mistakes. These pre-processing steps are performed on our dataset and it helps to increase the performance of our proposed model.

C. *Word Embedding*

The *Global log bi-linear* model is considered in this paper to avoid obfuscating the representation of words and apprehend semantic, syntactic, co-occurrence information for words, *Global log bi-linear* blend the benefits of the word2vec skip-gram model that prodigiously perform the task of word analogy with a latent semantic analysis that leverages statistical information.

The similarity between words is represented as indices in lexicon or vocabulary. Mikolov et al. [24] proposed Skimgram, which performs better on analogy tasks but fails to use corpus statistics since the model was not trained on global co-occurrence count. Pennington et al.[25] have proposed least-square Global vector model, the model is trained on the count of global word-word co-occurrence statistical information. The model produces a word vector with a significant substructure.

The GloVe vector model obtains meaning from word-to-word co-occurrence statistical information. The model constructs word-to-word co-occurrence (x_{ij}) sparse matrix M, where word j in the vocabulary occurs in the potential context of the word i. Let $x_{ij}=\Sigma_k x_{ik}$ represent the number time of word j occurs in the vocabulary. The probability of Word j occurs in word i is given by $P_{ij}=P(j|i)=x_{ij}/x_i$. The word vector learning is the ratio of co-occurrence probability; therefore, word learning is denoted as P_{ik}/P_{jk} where i, j, k are words.

D. *Composite Embedding*

The Glove vector represents a word vector with a significant substructure. To represent phrases in tweets, combine individual word vectors into composition embedding that represents a phrase of a tweet or an article. The composite embedding is achieved by Recursive Auto-

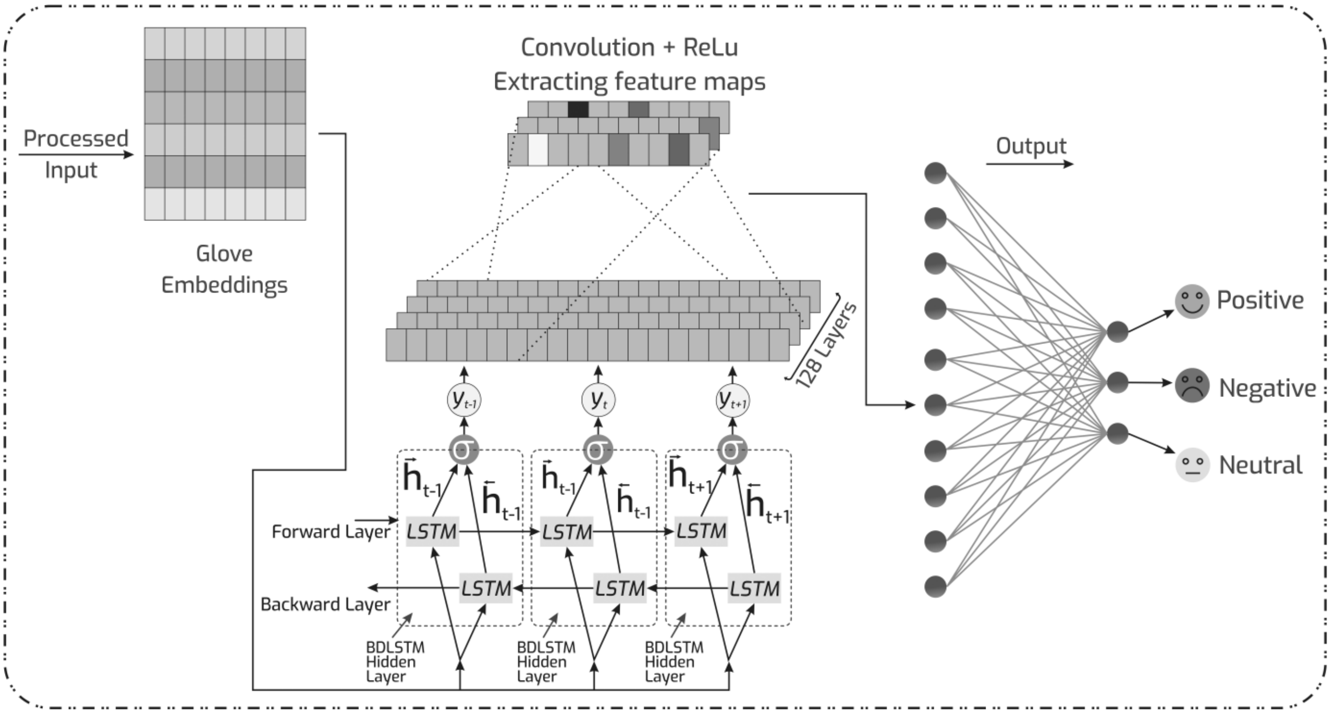


Fig. 4. Proposed hybrid CNN+LSTM model with Glove vector.

encoders (RAE)[23]. The composite embedding represents an entire tweet text sequence through continuous recursive parsing of words and sentence structure. It is good to use RNN-LSTM to parse the text of tweets into a sequence of words, with an order of the words that appear in the tweets. The RNN with LSTM is widely used to process a sequence of text without losing gradient.

E. *Convolutional Neural Network Model*

CNN has the ability to capture close semantic relationships in local regions of text. Still, LSTM can capture long-term semantic dependence between the words and sequences of words. Generally, the convolution layer of CNN performs feature extraction with the help of filters, and the non-linear features are extracted with the assistance of the ReLU activation function. The pooling layer reduces the feature map dimension, which helps compute computation cost in subsequent layers and effectively represents the feature of tweet text. A fully connected layer represents a structure between the features of tweets and the anticipated categories. Figure 4 represents the functionality flow between the distinct tier or phases of a CNN model and classifies tweets into three categories. The new set of tweets serves as an input to CNN. On completion of training, the final layer of CNN categorizes input text into distinct classes and assigns a probability for each category.

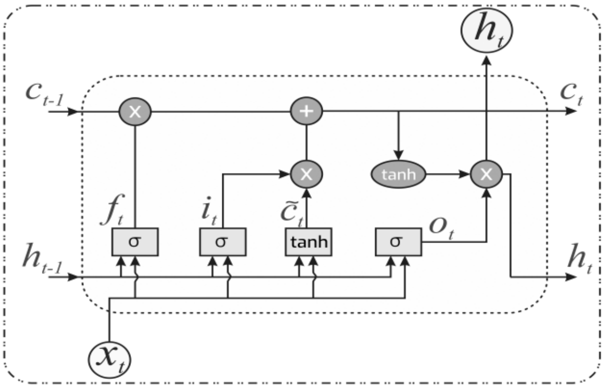


Fig. 5. LSTM Cell

F. *LSTM.*

Integration of CNN and LSTM results in an efficient sentiment classifier. CNN can extract a close semantic relationship between words, while LSTM performs well and solves long semantic dependence between words. In this research work, the tweets of the dataset are represented using a Global vector. Glove vector produces a word vector with a significant substructure, and it is based on global word-word co-occurrence statistical information. The Word vector produced by the Global vector is the CNN and LSTM model's inputs and the extracted features. Figure 4 illustrates the architecture of the proposed DistilBERT-CNN-LSTM model. The glove vector word embedding is input for CNN and LSTM models. The hybrid model can classify tweets according to sentiment by integrating the local dependence from CNN and long-term dependence from the LSTM model. LSTM is a popular and widely accepted RNN model. First, truncate the input sequence and perform padding to have all inputs with the same length. The embedded layer supports a vector length of 100 to represent each word. The next layer of LSTM has 100 memory units, and the output layer supports 13 output values. The softmax is used as the activation function.

The outputs of the feed-forward are considered as input to the neurons. The output at neuron depends on present input at time series *t*, and it establishes a relationship between two input values. In LSTM, input data is related to each other. The word embedding (*xt*) and word sentiment embedding (*st*) are combined as input to LSTM. Figure 4 shows the standard LSTM with *wt, Ot* as current the input and output, respectively at different time intervals *t*. Figure 5 illustrates the proposed LSTM model. The model has three essential switch: input switch *it*, output switch *Ot*, and forget switch *ft*. The input is entered through the input switch, and output is obtained through the output switch at time *t*. The comparison of present input with prior stature information (*h*(*t*−*1*)) and present input (*xt*) is decided at forget switch. Three gates decide the method of updating memory cell (*ci*) and current latency values ht. for a node in LSTM; mathematical relationships between these gates are shown below [26]:

$$i_t = \sigma(w_i.[h_{t-1}, x_t] + b_i) \quad (1)$$

$$f_t = \sigma(w_f\,.\,[h_{t-1}, x_t] + b_f) \quad (2)$$

$$o_t = \sigma(w_o\,.\,[h_{t-1}, x_t] + b_o) \quad (3)$$

$$c_i = tanh(w_c\,.\,[h_{t-1}, x_t] + b_c) \quad (4)$$

$$c_i = f_t * c_{t-1} * i_t * c_t \quad (5)$$

$$h_t = o_t * tanh(c_t) \quad (6)$$

The LSTM processes words from word vectors. It parses whole tweets and determines the similarity and long-term relationship between words from the beginning to the end of tweets. CNN model identifies the short-term relationships between words from the beginning to the end of the tweet. In the proposed work, CNN and LSTM models are used for the experiments.

V.EXPERIMENT RESULTS AND ANALYSIS

The proposed hybrid CNN model results with the GloVe word embedding is compared with well-known machine learning algorithms such as Random Forest, Support Vector Machine, Multinomial Nave Bayes K-Nearest Neighbours (KNN), and XG Boost. All these machine learning algorithms are known for sentiment analysis. We considered 10007 English tweets specific to the 2018 FIFA world cup for our experiment. All tweets of the dataset are annotated manually.

We carried out experiments on Intel i5, 32GB RAM, GTX 1080 Ti graphic card, and Python 3.5 are used as the development programming language. The Scikit-learn Tensorflow Python libraries are used while developing the proposed model. We evaluated the performance of the proposed model with various performance evaluation metrics such as precision, recall, F1 score, validation accuracy, testing accuracy, sensitivity, specificity, and validation loss to assess the performance of the proposed model. The proposed DistilBERT-CNN-LSTM model is executed 20 times and obtained classification results.

TABLE I
PARAMETERS IN PROPOSED DistilBERT-CNN-LSTM MODEL

Layer Name	Layer	Input Shape	Output Shape	Param.
Proc. Input1	Input	(None, None)	(None, None)	0
$Tokenize_1$	Tokenizer	(None, None)	(None, None)	0
Zero Pad_1	Pad sequences	(None, None)	(None, 51)	0
Proc. $Input_2$	Input	(None, 51)	(None, 51)	0
$GloVe_1$	Embedding	(None,51)	(None,51,300)	1621800
$Bidir._1$	LSTM	(None,51,300)	(None,51,128)	186880
$Conv_1$	Conv. 1D	(None, 51,128)	(None,48,10)	5130
$Flat_1$	Flatten	(None,48,10)	(None,480)	0
$Drop_1$	Dropout	(None,480)	(None, 480)	0
$Linear_1$	Dense	(None,480)	(None,3)	1443

Figure 6 depicts the results of training, testing classification accuracy for the proposed DistilBERT-CNN-LSTM model. The proposed model achieves an accuracy of about 85% in tweets' classification into positive, negative, and neutral based on usage CNN LSTM with GloVe word embedding. The proposed model extract sentiment information, short and long-term dependence between the words of tweets. The GloVe construct has high-quality Word embedding for a small and large set of unlabeled, incomplete tweets with the help of co-occurrence and can confiscate semantic information of words in tweets.

GloVe constructs word embedding with the help of co-occurrence and semantic information of words in tweets for incomplete sentences. Figure 7 illustrates the loss in the classification of tweets. It is observed in figure 7 that loss has increased slightly because of incomplete sentences in the dataset.

Predictions of the proposed model
Tweet: FrenchcommentatorsreactiontoPavardgoalispriceless
Original sentiment: Positive
Predicted sentiment: Positive

Misclassification by the model
Tweet: *Can we all agree that Pavards Goal for France is the Goal of the*
Original sentiment: Positive
Predicted sentiment: Neutral

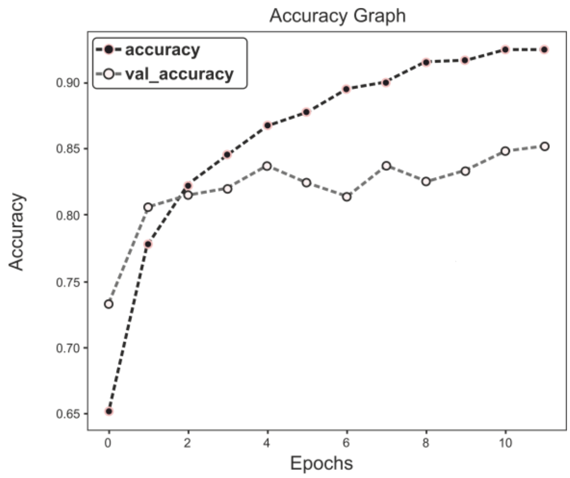


Fig. 6. Classification Accuracy of DistilBERT-CNN-LSTM model

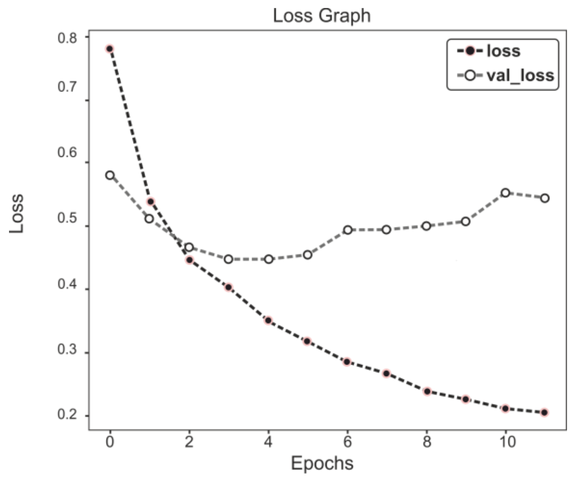


Fig. 7. Classification Loss of DistilBERT-CNN-LSTM model

TABLE II
EVALUATION OF PERFORMANCE PARAMETERS

Category	Precision	Recall	F1-Score	Support
Positive	0.90	0.89	0.89	560
Negative	0.78	0.63	0.70	115
Neutral	0.78	0.83	0.81	326
			Accuracy - 0.85	

TABLE III
EXPERIMENTAL RESULTS OF TRAINING AND VALIDATION METRICS

Metrics	Training	Validation
Final Loss	0.2059	0.5458
Final Accuracy	0.9249	0.8520
Precision	0.9313	0.8574
Sensitivity @Specificity	0.9990	0.9830
Specificity @Sensitivity	0.9995	0.9920

TABLE IV
COMPARISON OF PROPOSED WORK WITH M L ALGORITHMS

Model Name	Training Accuracy	Validation Accuracy
Our Work:Hybrid LSTM-CNN	0.92	0.85
Random Forests	0.95	0.82
Support Vector Machine	0.94	0.81
Multinomial Nave Bayes	0.83	0.76
K Nearest Neighbours (KNN)	0.80	0.75
XG Boost	0.72	0.68
Logistic Regression	0.88	0.81

Table 2 shows the Precision, Recall, and F1 score of the proposed model with the GloVe embedding approach. The proposed model classifies tweets as positive, negative, and neutral. The proposed model with GloVe word vector blend advantage of word2vec, the skip-gram model with matrix factorization, explores global statistics and sentiment information. It is observed from Table 2 that the Precision, Recall, and F1 score of the word vector representation used in this paper is better than other word representation methods such as TF-IDF and Word2Vec. The proposed DistilBERT-CNN-LSTM with the GloVe word vector approach has increased the precision by 2.44 %, the recall by 4.73 %, and the F1 score by 3.58 %. The steady increase in precision, recall, and F1 score is due to the usage of GloVe, which captures the semantic and sentiment information of tweets. It also reflects the importance and frequency of different words in tweets. The efficient word vectors are input into the DistilBERT-CNN-LSTM model to identify and analyze sentiments in tweets.

Table 3 represents the final training, validation accuracy, precision, sensitivity, and specificity of experimental values. The proposed DistilBERT-CNN-LSTM model can effectively capture and transfer the semantics information between words of tweets. The CNN captures fine-grained local information, LSTM alleviates gradient disappearance and builds a long-term relationship between words in lexicons. The word representation by the GloVe method leverages the sentiment information present in the words of tweets and contributes to the sentiment analysis and classification of tweets.

The experiment results of the proposed model and prevalent machine learning algorithms are given in Table 4. The proposed hybrid model with the GloVe word embedding approach can identify and classify tweets' sentiment with an accuracy of 85%, while the Random Forest classifier achieves an accuracy of 82%. The accuracy difference between RF and Logistic Regression (RL) is not significant. The machine learning algorithms use the most common word embedding techniques that do not utilize the co-occurrence of the word-word and capture semantic and syntactic information. The performance of other ML classifiers a steady decrease because they use general lexicons. The Football specific lexicon with a public lexicon will improve classification accuracy.

Figure 8 and Figure 9 demonstrate the performance of machine learning algorithms. Figure 8 and Figure 9 give the comparison of different classifiers; it is observed that Random Forest is accurate with about 84% accuracy, Linear SVM and Multinomial Naive Bayes provide about 74%, and Logistic Regression gives high performance with 86% accuracy. F_1 score and macro-average F1, and weighted average are shown in the figure 8.

We have observed the effect of blending different features on multiple classifiers' performance from experiment results. Analyzing the classifier's performance with the general lexicon feature, the fusion of the domain-specific lexicon features has improved the performance of the sentiment classifiers. Blending general and domain-specific lexicon features have improved SVM, MNB, and RF performance in determining sentiment in tweets related to football events.

The Random Forest (RF) and Multinomial Naïve Bays (MNB) classifier achieve 80-82% accuracy using uni-gram without stop words. The performance of Gradient Boosting (XGB) achieves lower accuracy using uni-gram.

We investigated the performance of all classifiers for average words size and maximum words size.

The use of appropriate word embedding will guarantee correct sentiment identifications in the tweets. It is observed from figure 10 that uni-gram and bag-of-words (BGW) are insufficient in finding sentiments because they discard the order of words and context of the text. Since GloVector improves learning sentiments in tweeter by considering the context of the text, short and long-term dependency of text, the accuracy in finding sentiments in tweets is better than uni-gram and bag-of-word word embedding. The domain and linguistic lexicons have improved the learning sentiments over uni-gram, bag-of-words word embedding techniques.

For experimental analysis, we considered 10007 tweets and constructed a lexicon of 4512 words. The length of tweets varies from 6 to 70 words with stop words and 2 to 60 words without stop words. The experiments are carried out with words sizes of 10, 20, 30, 40, 50 and 60 on XGB, Logistic Regression and Proposed LSTM-CNN classifier using uni-gram, bag-of-words and GloVector. Figure 10(a) illustrates the impact of words size with and without stop words in learning sentiments.

The stop words have a significant role in finding sentiment in the proposed DistilBERT-CNN-LSTM classifier. However, uni-gram and bag-of-words-based models perform better without stop words and their overall performance is not better than the proposed DistilBERT-CNN-LSTM classifier.

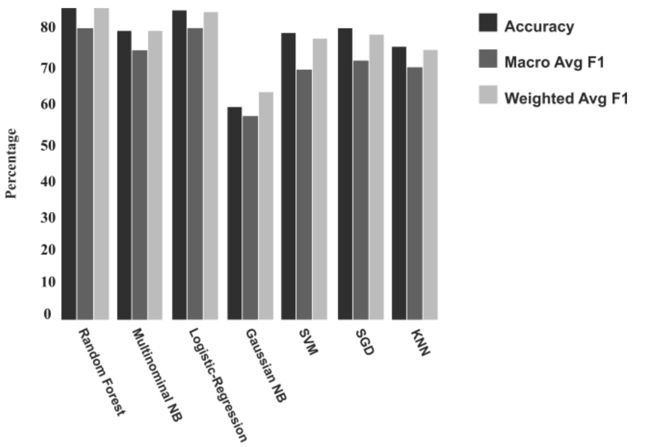


Fig. 8. Accuracy, macro average F1, and weighted average F1 using different classifier models.

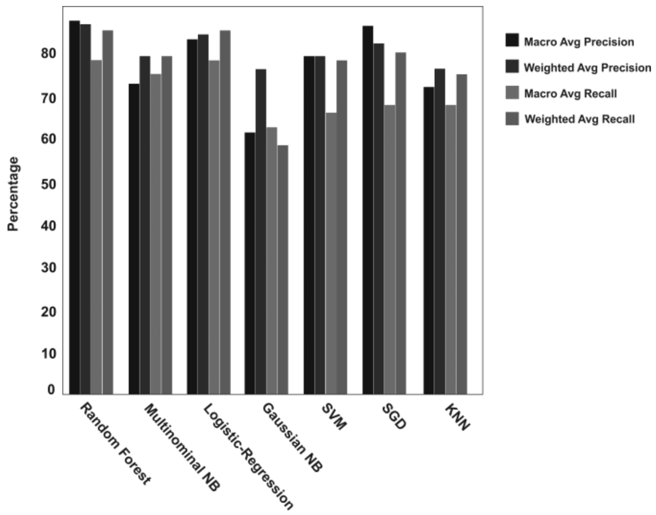
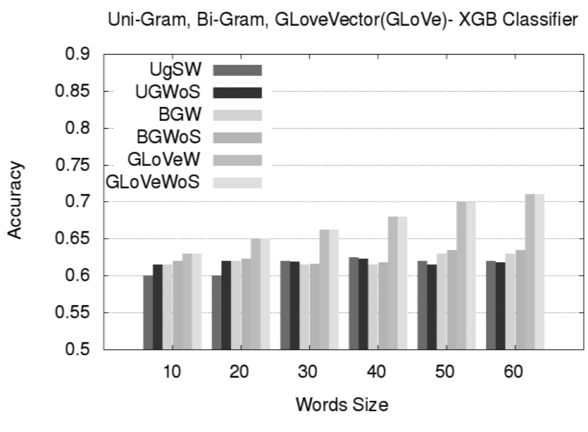
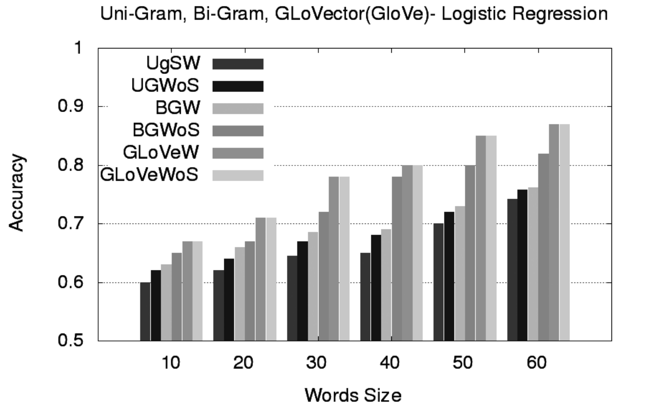


Fig. 9 The performance of the different classifiers with respect to Macro Average Precision, weighted average Precision, macro-average Recall, and weighted average.

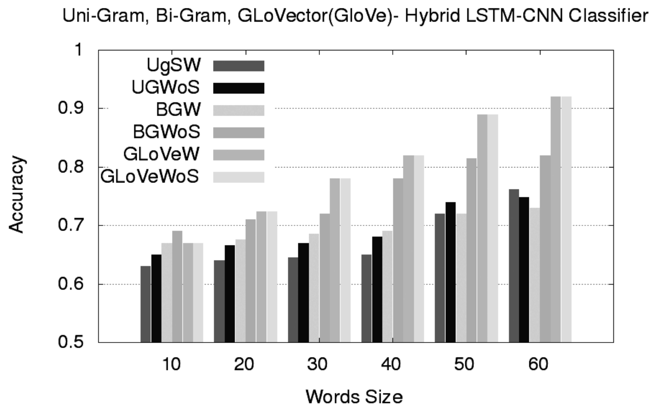


A. Performance of XGB Classifier with Uni-Gram, Bi-Gram and GloVector with or without stop words.

Figure 10(b) shows that logistic regression (LR) has better performance when words size is about 50 to 60 using GloVector word embedding. Similarly, Gradient Boost (XGB) classifier has good performance with GloVector with and without stop words at words size is about 40-60. It is observed from figure 10(c) that uni-gram and bag-of-words-based classifier performance is not encouraging and fails to identify sentiments in tweets due to a lack of context-text order of words.



B. *Performance of Logistic Regression Classifier with Uni-Gram, Bi-Gram and GloVector with or without stop words.*



C. *Performance of LSTM-CNN Classifier with Uni-Gram, Bi-Gram and GloVector with or without stop words.*

Fig. 10. Performance of different Classifiers with Uni-Gram, Bi-Gram, and GloVector.

Figure 11. shows words with hashtags showing some sign of sentiments that would augment sentiments analysis. It is observed from figure 11 that the results of micro average F-score for LR, MNB, SVM, and RF classifier is about 65-55% with the hashtag and 60-50% without the hashtag, the micro average F-score is about 75-85 % for proposed LSTM-CNN classifier. The proposed LSTM-CNN classifier using hashtags might contain meager sentiment facts that act as ancillary along with GloVector to learn sentiments. The LSTM-CNN classifier yields a micro average F-score of 77% with words containing hashtags and 80% with words excluding hashtags.

The classifier with a domain-specific lexicon does not perform well. However, combining domain and generic lexicon with the GloVector word embedding will improve performance by 5-10 % than a classifier trained on domain-specific lexicon alone. Figure 12 shows the performance of classifiers with domain-specific lexicons. The combination of domain lexicon and generic lexicon will enhance the sentiment learning process. The competence of the proposed DistilBERT-CNN-LSTM model improves as the number of epochs increases. Selecting the correct number of epochs plays a vital in the performance of the model. If the number of epochs is too large then it causes an overfitting problem.

TABLE V: COMPARISON OF PROPOSED MODEL PERFORMANCE WITH STATE-OF-ART LANGUAGE TRANSLATION MODELS

Hybrid Model Name	**XLNET**						**XLNET + CNN**					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.000	0.538	0.000	0.538	0.200	0.230	0.200	0.578	0.210	0.542	0.310	0.330
Recall	0.000	1.000	0.000		0.300	0.320	0.240	1.000	0.200		0.420	0.432
F1 Score	0.000	0.700	0.000		0.260	0.280	0.300	0.780	0.000		0.320	0.347
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												
Hybrid Model Name	**XLNET + CNN + LSTM**						**DistilBERT**					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.330	0.638	0.300	0.581	0.400	0.430	0.751	0.880	0.835	0.856	0.823	0.856
Recall	0.320	1.000	0.350		0.430	0.473	0.770	0.905	0.801		0.823	0.846
F1 Score	0.310	0.720	0.500		0.496	0.505	0.760	0.900	0.813		0.820	0.846
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												
Hybrid Model Name	**DistilBERT + CNN**						**DistilBERT + CNN + LSTM**					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.753	0.888	0.839	0.859	0.827	0.859	0.761	0.895	0.850	**0.865**	**0.836**	0.865
Recall	0.775	0.909	0.808		0.826	0.848	0.790	0.915	0.812		0.839	0.865
F1 Score	0.764	0.902	0.816		0.822	0.848	0.776	0.905	0.831		0.837	0.865
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												
Hybrid Model Name	**ALBERT**						**ALBERT+CNN**					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.699	0.873	0.848	0.845	0.807	0.846	0.706	0.877	0.856	0.852	0.817	0.853
Recall	0.752	0.905	0.782		0.813	0.845	0.763	0.915	0.804		0.825	0.858
F1 Score	0.725	0.889	0.813		0.809	0.845	0.732	0.899	0.823		0.816	0.858
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												

Hybrid Model Name	ALBERT + CNN + LSTM						BERT					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.716	0.883	0.862	0.858	0.827	0.873	0.719	0.889	0.854	0.858	0.821	0.859
Recall	0.781	0.924	0.812		**0.866**	0.867	0.781	0.911	0.801		0.831	0.858
F1 Score	0.747	0.911	0.846		0.834	0.876	0.749	0.900	0.827		0.825	0.858
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												
Hybrid Model Name	**BERT + CNN**						**BERT + CNN + LSTM**					
Classification	a	b	c	d	e	f	a	b	c	d	e	f
Precision	0.735	0.896	0.867	0.858	0.829	0.875	0.743	0.901	0.210	0.860	0.832	**0.887**
Recall	0.801	0.924	0.822		0.841	0.865	0.810	0.932	0.200		0.854	**0.876**
F1 Score	0.764	0.912	0.836		0.832	0.864	0.776	0.923	0.000		**0.853**	**0.886**
Support	105	539	357	1001	1001	1001	105	539	357	1001	1001	1001
a: Negative, b: Positive, c: Neutral, d: Accuracy, e: Macro Average, f: Weighted Average												

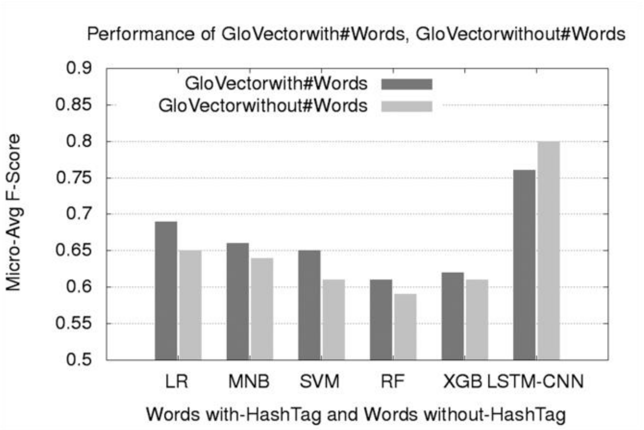


Fig. 11 Performance of GoVector with/without #Tags.

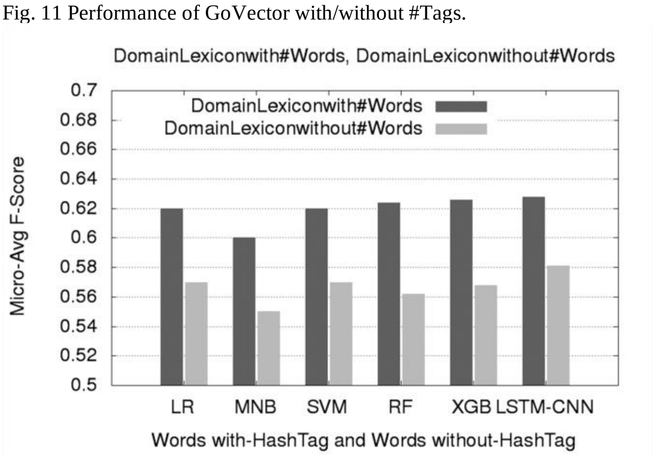


Fig. 12 Performance of classifier using Domain Lexicon with/without #Tags.

The classification accuracy of the proposed DistilBERT-CNN-LSTM model is increasing gradually with the rise in the number of epochs. The performance of the model is almost constant after the 40th epoch. The performance of the model is about 92% for the 120th epoch. Figure 13 demonstrates the performance of the proposed model at a different number of epochs.

The model training process is long and sometimes it gets stuck if the learning rate of the model is limited. The performance of the model is substandard and produces an ambiguous set of weights if the learning rate of the model is large.

Figure 14 shows the effects of the learning rate on the performance of the proposed DistilBERT-CNN-LSTM model. The accuracy of the model is about 92 % when the learning rate is between 0.1 and 0.3; it is because the training process is accelerated. The accuracy of the model decreases with an increase in the learning rate. The small variation in the learning rate requires fewer epochs.

The average results of repeated experiments for different word vector representations are drawn in Figure 15. The proposed model uses Glove Vector to find sentiment information, semantic information, and short or long-term relationship between the words. The proposed model uses GloVector as a word embedding method because it learns sentiments in tweeter by considering the context of the text, short and long-term dependency of text. TF-IDF finds the distribution of words belonging to different classes but fails to learn the sentiment between words at a short distance or longer distances. The Seninfo finds sentiment information associated with words but discards the short or long-term relationship between words. Senifo+TFIDF finds the distribution of words and sentiment information but fails to capture semantic information in words. Hence, the accuracy of the proposed model is above 92%.

Figure 16 shows the performance of the model with a weight of the word. When the weight of a word is 1 then accuracy of the proposed model is about 90.87%. The value of accuracy is exponentially increased and then decreases gradually as the value of the weight of the word increases because the proposed model captures the sentiment and semantic information associated with words in the tweets. More words with sentiment and semantic information are available in the initial stage. Hence, the F1 score is high. In the later stage of the experiment, the non-sentiment word in the tweets is more. Hence, the F1 score decreases.

Table 6 shows the performance of the hybrid model with different language models. The results of ALBERT are compared to the original BERT, and the simulation results reveal that the proposed model DistilBERT+LSTM+CNN performs substantially better than all other models.

DistillBERT is lighter and the language understanding capability of DistillBERT is faster than BERT. Knowledge distillation is a refining approach in which a smaller model - the student - is trained to recreate the performance of a larger model - the teacher. DistilBERT is trained on a triple loss with student initialization.

BERT is a mask language model that pre-trains unlabeled data using deep bidirectional representations. The goal of the masked language model (MLM) is to predict the original vocabulary id of the masked tokens from the input at random. BERT was designed in such a way that it replaces random words in phrases with a specific [MASK] token and attempts to guess what the original word was. BERT forgot to replace [MASK] tokens at the end of pre-training and it ignores the dependence between the masked places. This work use XLNet a generalized pre-training method that allows bidirectional context learning by maximizing the expected likelihood across all possible factorization order permutations.

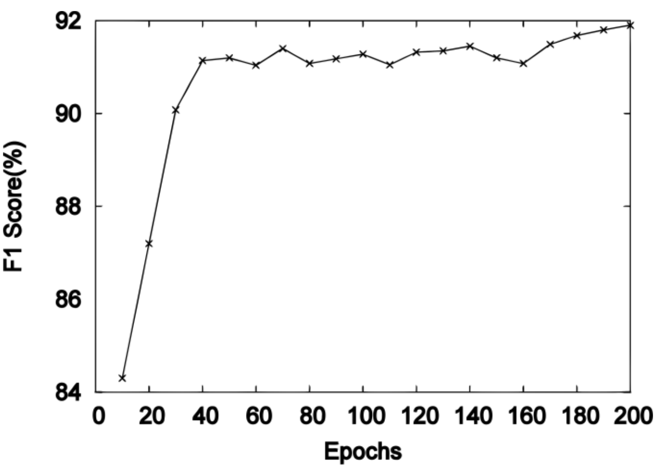


Fig. 13. Model's Accuracy at different numbers of Epochs.

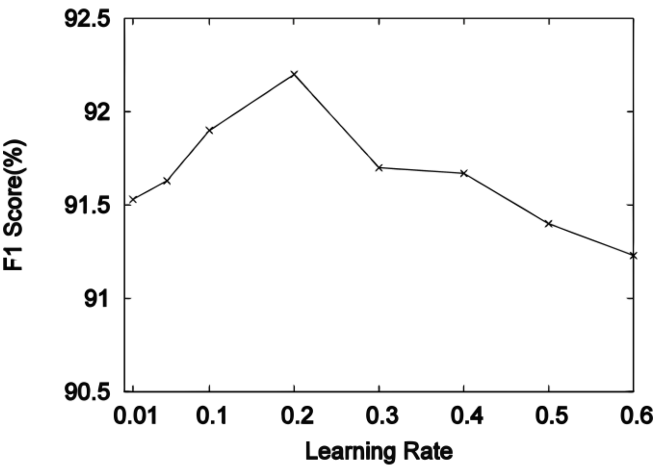


Fig. 14 Learning Rate of DistilBERT-CNN-LSTM model

ALBERT uses self-supervised language representation learning for language representation learning. ALBERT solves the problem of high memory usage by implementing multiple memory reduction strategies, such as Factorized Embedding Parameterization, cross-layer parameter sharing, and Sentence Ordering Objectives.

The proposed hybrid model DistilBERT+LSTM+CNN reduces the model size by 40%, it has 97% better language understanding capability, and is 60% faster than other models.

Table 5 illustrates comprehensive comparison results of proposed model performance with different state-of-art models on performance metrics such accuracy, true positive, true negative rate, F1 score, macro average, and weighted average.

TABLE VI
COMPARISON OF PROPOSED HYBRID MODEL WITH LANGUAGE REPRESENTATION MODELS

Hybrid Model Name	Classification	Preci.	Recall	F1-SCore	Accur.
XLNet+ LSTM+ CNN	Negative	0.000	0.000	0.000	0.538
	Positive	0.538	1.000	0.700	
	Neutral	0.000	0.000	0.000	
DistilBERT +LSTM +CNN	Negative	0.761	0.790	0.776	**0.865**
	Positive	0.895	0.915	0.905	
	Neutral	0.850	0.812	0.831	
ALBERT+ LSTM+ CNN	Negative	0.699	0.752	0.725	0.845
	Positive	0.873	0.905	0.889	
	Neutral	0.848	0.782	0.813	
BERT+ LSTM+ CNN	Negative	0.719	0.781	0.749	0.858
	Positive	0.889	0.911	0.900	
	Neutral	0.854	0.801	0.827	

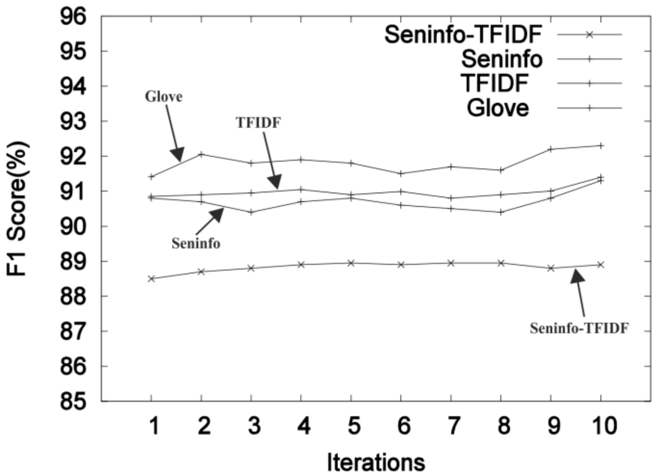


Fig. 15 Accuracy of proposed DistilBERT-CNN-LSTM model at different word representation methods.

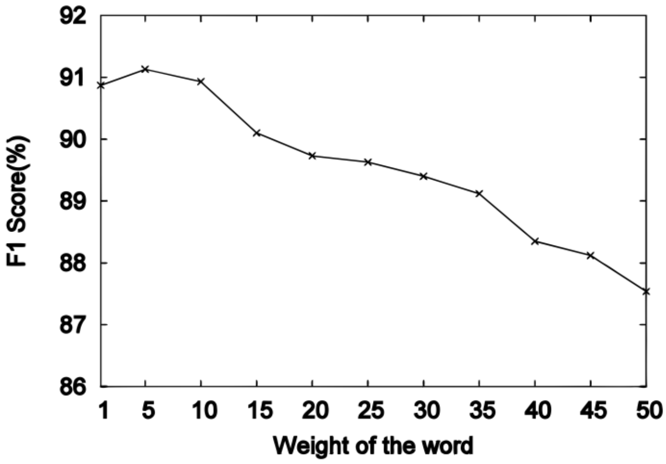


Fig. 16 Accuracy of proposed DistilBERT-CNN-LSTM model with different weights of the word.

The performance of the proposed model across all sets of the dataset is evaluated through the macro average of precision. The macro average reflects the significance of all classes in an unbalanced dataset. The weighted average gives more credits to classes with higher instances in the dataset. It is noticed from table 6 that BERT+CNN+LSTM has an 85.3 % macro average and 88.6 % weighted average. The DistillBERT is an extended version of BERT. It is the bidirectional model which reads whole words sequence and learns a word's context by looking at its left and right surroundings. The proposed model DistilBERT+LSTM +CNN achieved 86.5% accuracy and 83.6 % macro average precision since the proposed model has better generalization, language understanding potential, and performs downstream tasks efficiently.

The predicting inter-sentence semantic continuity helps in downstream tasks. As a result, the proposed model performs better than state-of-art models. It is easy to pre-train the proposed model since it is faster, lighter, and smaller than state-of-art models. BERT+CNN+LSTM has 88.7%, 87.6%, and 88.6 % weighted average of precision, recall, and F1 scores. The extended version of BERT+CNN+LSTM is DistilBERT+LSTM+CNN which uses next sentence prediction and masked language strategies to predict the next word.

VI. CONCLUSIONS

The traditional Word embedding technique leverage semantic information but fails to capture the sentiment of words. With GloVe word embedding approach, the sentiment embedding of a word is created. The word vector with semantic and sentiment embedding is fed as input to the proposed DistilBERT-CNN-LSTM model that performs sentiment analysis. We have used 10007 tweets specific to the 2018 FIFA world cup and tweets are manually annotated for our experiment. The experiment results show that the blended CNN and LSTM model with the GloVe word embedding approach provides a robust and comprehensive and robust model for sentiment classification.

ACKNOWLEDGMENT

The authors would like to thank all my colleagues and research scholars of the department for their valuable comments.

REFERENCES

[1] X. Fu, J. Yang, J. Li, M. Fang, and H. Wang, "Lexicon-Enhanced LSTM with Attention for General Sentiment Analysis ", IEEE Access, vol. 6, pp. 71884-71891, 2018, doi: 10.1109/ACCESS.2018.2878425.

[2] H. T. Phan, V. C. Tran, N. T. Nguyen and D. Hwang, "Improving the Performance of Sentiment Analysis of Tweets Containing Fuzzy Sentiment Using the Feature Ensemble Model", IEEE Access, vol. 8, pp. 14630-14641, 2020, doi: 10.1109/ACCESS.2019.2963702.

[3] P. Barnaghi, P. Ghaffari, and J. G. Breslin, "Text Analysis and Sentiment Polarity on FIFA World Cup 2014 Tweet", In Proceeding of ACM Conference on SIGKDD, vol. 15, pp. 10-13, 2015.

[4] P. Barnaghi, P. Ghaffari and J. G. Breslin, "Opinion Mining and Sentiment Polarity on Twitter and Correlation between Events and Sentiment", In Proceedings of IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService), Oxford, pp. 52-57, 2016, doi: 10.1109/BigDataService.2016.36.

[5] Salur, Mehmet Umut and Aydin, Ilhan, "A Novel Hybrid Deep Learning Model for Sentiment Classification", IEEE Access, vol. 8, pp. 58080-58093, April 2020.

[6] Xu, Guixian and Meng, Yueting and Qiu, Xiaoyu and Yu, Ziheng and Wu, Xu, "Sentiment Analysis of Comment Texts based on BiLSTM", IEEE Access, vol. 7, pp. 51522-51532, April 2019.

[7] Saad, Shihab Elbagir and Yang, Jing, "Twitter Sentiment Analysis based on Ordinal Regression", IEEE Access, vol. 7, pp. 163677-163685, November 2019.

[8] Zhang, Bowen and Xu, Xiaofei and Yang, Min and Chen, Xiaojun and Ye, Yunming, "Cross-domain Sentiment Classification by Capsule Network with Semantic Rules", IEEE Access, vol. 6, pp. 58284-58294, October 2018.

[9] Aydin, Cem Rifki and Gungor, Tunga, "Combination of Recursive and Recurrent Neural Networks for Aspect-Based Sentiment Analysis Using Inter-Aspect Relations", IEEE Access, vol. 8, pp. 77820-77832, May 2020.

[10] Phan, Huyen Trang and Tran, Van Cuong and Nguyen, Ngoc Thanh and Hwang, Dosam "Improving the Performance of Sentiment Analysis of Tweets Containing Fuzzy Sentiment using the Feature Ensemble Model", IEEE Access, vol. 8, pp. 14630-14641, January 2020.

[11] Shayaa, Shahid and Jaafar, Noor Ismawati and Bahri, Shamshul and Sulaiman, Ainin and Wai, Phoong Seuk and Chung, Yeong Wai and Piprani, Arsalan Zahid and Al-Garadi, Mohammed Ali, "Sentiment Analysis of Big Data: Methods, Applications, and Open Challenges", IEEE Access, vol. 6, pp. 37807-37827, July 2018.

[12] Fatimah Alzamzami, Mohamad Hoda, Abdulmotaleb El Saddik, "Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation", IEEE Access, vol. 8, pp. 101840-101858, 2020, doi:10.1109/ACCESS.2020.2997330

[13] W. Xing and Y. Bei, "Medical Health Big Data Classification Based on KNN Classification Algorithm", IEEE Access, vol. 8, pp. 28808-28819, 2020, doi: 10.1109/ACCESS.2019.2955754.
[14] Jonathan Gratch, G.Lucas, N.Malandrakis, E.Szablowski, E.Fessler, and J.Nichols, "GOAALLL!: Using Sentiment in the World Cup to Explore Theories of Emotion", In Proceedings of International Conference on Affective Computing and Intelligent Interaction, pp. 898-903, 2015.
[15] S. Rosenthal, A. Ritter, P. Nakov, and V. Stoyanov, "Semeval-2014 Task 9: Sentiment Analysis in Twitter", In Proceedings of the 8^{th} International Workshop on Semantic Evaluation (SemEval 2014), pp.73-80, 2014.
[16] S. Aloufi, F. Alzamzami, M. Hoda, and A. E. Saddik, "Soccer Fans Sentiment Through the eye of Big Data: The UEFA Champions League as A Case Study", In Proceedings of IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), pp. 244-250, 2018.
[17] Rui Wang, Deyu Zhou, Mingmin Jiang, Jiasheng Si, Yang yang, "A Survey on Opinion Mining: From Stance to Product Aspect", IEEE Access, vol. 7, pp. 41101-41124, 2019.
[18] Unggul Widodo Wijayanto, Riyanarto Sarno, "An Experimental Study of Supervised Sentiment Analysis Using Gaussian Nave Bayes", In Proceedings of International Seminar on Application for Technology of Information and Communication, pp. 476-481, 2018.
[19] Samah Aloufi, Abdulmotaleb El Saddik, "Sentiment Identification in Football-Specific Tweets", IEEE Access, vol.6, pp. 78609-78621, 2018.
[20] Siti Nurmaini, Alexander Edo Tondas, Radiyati Umi Partan, Muhammad Naufal Rachmatullah, Annisa Darmawahyuni, Firdaus Firdaus, Bambang Tutuko, Rachmat Hidayat, and Ade Iriani Sapitri, "Automated Detection of COVID-19 Infected Lesion on Computed Tomography Images Using Faster-RCNNs," Engineering Letters, vol. 28, no.4, pp1295-1301, 2020.
[21] Qinghe Zheng, Mingqiang Yang, Xinyu Tian, Xiaochen Wang, and Deqiang Wang, "Rethinking the Role of Activation Functions in Deep Convolutional Neural Networks for Image Classification," Engineering Letters, vol. 28, no.1, pp. 80-92, 2020.
[22] Caiwen Wang, and Youlong Yang, "Nearest Neighbor with Double Neighborhoods Algorithm for Imbalanced Classification," IAENG International Journal of Applied Mathematics, vol. 50, no.1, pp. 147-159, 2020.
[23] Socher, Richard and Pennington, Jeffrey and Huang, Eric H and Ng, Andrew Y and Manning, Christopher D, " Semi-supervised Recursive Autoencoders for Predicting Sentiment Distributions ", p*roceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 151-161, 2011.
[24] Mikolov, Tomas and Chen, Kai and Corrado, Greg and Dean, Jeffrey, "Efficient Estimation of Word Representations in Vector Space", *arXiv preprint arXiv:1301.3781*, 2013.
[25] Pennington, Jeffrey and Socher, Richard and Manning, Christopher D "Glove: Global Vectors for Word Representation", *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532-1543, Jun. 2014.
[26] Cui, Zhiyong and Ke, Ruimin and Pu, Ziyuan and Wang, Yinhai "Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-wide Traffic Speed Prediction", *arXiv preprint arXiv:1801.02143*, 2018.

Venkatesh (M'18) is currently working as an Assistant Professor in the Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore-560001. Obtained his Bachelor of Engineering, Masters of Technology, and Ph.D. in Computer Science and Engineering in 2000, 2004, and 2018 respectively. His research interest includes Wireless Sensor Network, Ad-hoc Network, Data Science, and Web Recommendation System.

Siddhanth U Hegde is a B.Tech student of the Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bengaluru-560001, INDIA (e-mail: siddhanthhegde227@gmail.com).

Satish B Basapur is currently an Assistant Professor in the Department of Information Science and Engineering, Dr. Ambedkar Institute of Technology, Bangalore-560056.

Nagaraju Y, Research Scholar, Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore-56001. He obtained his Bachelor of Engineering degree in Information Science and Engineering and Masters of Engineering degree in Information Technology in 2015 and 2017. His research interest includes Deep Learning and Computer Vision.