LSTM-based Deep Learning Model for Emotion Intensity Level by Enhanced Sentiment Classification

Panthagani Vijaya Babu[1], T.R.Rajesh[2], Anjaneyulu Nelluru[3]

[1,2,3]Assistant Professor

[1,2,3]Department of CSE

[1,2,3]Vignan's Foundation for Science, Technology & Research (Deemed to be University),

Guntur, Andhra Pradesh, India

[1]pathagani.vijay@gmail.com, [2]trrajesh542@gmail.com, [3]nelluri.anjaneyulu5@gmail.com

Abstract

In last decades, social networking sites like Twitter and Facebook has provided a platform to express people opinions about product, politics, sports, entertainment and so on. At the same time, emotion and sentiment classification problem also attracted several researchers to analyze the state of mind of the people, which finds useful to improve the quality of the product or services. The recent developments of deep learning (DL) techniques started to be used for emotion and sentiment classification of social networking data. This paper presents an effective DL based Long Short-Term Memory (LSTM) model for emotions with intensity level sentiment classification called LSTM-EISC for Twitter text. The proposed LSTM-EISC model follows a two-step procedure, namely preprocessing and classification. The LSTM model is applied to classify multiple intensities of Twitter data such as Fear, Anger, Joy, and Sadness. A series of experiments takes place on SEMEVAL2018 Task-1Emotion Intensity Ordinal Classification dataset to ensure the effective performance of the LSTM-EISC model. The simulation outcome pointed out the effective classification results of the LSTM-EISC model over the existing methods under several aspects.

Keywords: *Deep Learning, Tokenization, Sentiment Analysis, Data Classification, Twitter.*

1. Introduction

Sentiment analysis (SA) is mainly used to analyze the suggestions, emotions, appraisals, attitudes, behavior and attributes represented in a text [1]. Using the progressive development of social networking on the Internet like reviews, forum comments, blogs and news, many users can exchange the views and opinions through online. In line with, it is considered as fascinating issue among business people and public. Different sentiments expressed by people in Twitter about Amazon, Google and Facebook. The major dimension of SA is sentence-level SA. The traditional works mainly concentrates on finding the polarity of a sentence such as positive, negative, and neutral which depends upon the language clues obtained from textual sentences [2]. This problem was considered as a typical one with no assumption of diverse types of sentences. Therefore, various sentences represent the emotions in diverse manner. For instance, a sentence "It is good.", the sentiment polarity is surely positive; for an interrogative sentence like "Is it good?", the sentiment polarity is ambiguous, and it is comparatively referred as negative; and for relative sentence "A is better than B.", here sentiment polarity could not be decided, due to the dependency of opinion target which aims on (A or B) [3].

On the contrary factual text, sentiment content can be represented in a random fashion, which becomes very complex to find every constituent. It is defined that, to obtain 1-technique-fits-all solution and a divide-and-conquer model is required for solving the special sentences using exclusive features, which is said to be diverse types of sentences has to be treated with similar sentence-level SA. The SA model applies several modules to perform the sentence classification. Sentences are classified into subjective as well as objective that divides the opinions from actual facts. Only few developers concentrated on target-dependent sentiment classification that is applied to classify a sentiment polarity for the target on sentences with explicit sentiment while some of the mining opinions in relative sentences that is to compute the degree of positivity [4]. In addition, the SA focused on conditional sentences, sentences with modality, that is composed with specialized features which results in a complex processing in sentiment orientations [5].

There are massive works which has been employed with Machine Learning (ML) approaches for the purpose of text categorization. Even though ML models were applied extensively and exhibited optimal function, and depends upon manually described features, in which feature description needs maximum effort to domain professionals. Additionally, DL approaches acquire maximum attention as it is helpful in reducing the effort for feature definition and attains maximum performance. Here, the main aim of the sentiment classification is to classify text data and present the structure of LSTM, which belongs to DL method.

Numerous works are presented to classify the sentiments under the application of ML schemes, like Support Vector Machine (SVM), Naive Bayes (NB), Maximum Entropy (ME), Stochastic Gradient Descent (SGD), as well as ensemble are the integration of 2 convolutional layers and 2 pooling layers for classifying tweet data of 4 languages in a sentiment fashion and reaches better F1-Score. [6] projected that, CNN approach is composed with 3 convolution/pooling layer pairs, and performs well when compared with existing techniques such as Matrix-Vector Recursive Neural Network (MV-RNN) [7]. Some of the experiments were processed with the application of CNN approaches which is composed with diverse structures [8].

Though the ML model attains effective performance with the employment of handcrafted features and n-gram features, it is still comprised with few shortcomings with diverse features; and in various data, maximum effort of professionals would be required to reach good performance. These shortcomings are involved in fusion models for SA which integrates alternate resources like ontology, lexicon, as it costlier as well as effort of domain professionals. DL approaches are considered as one of the solutions for these shortcomings, and referred to apply in capturing random patterns in an automated manner. Therefore, as revealed in [9], under the application in DL approaches for SA would provide a meta-level feature presentation which normalizes well on novel applications.

The DL and ML approaches are currently applied for sentiment classification. [10] developed a novel approach, named as Adaptive Recursive Neural Network (AdaRNN), which classifies Twitter data into 3 sentiment labels namely, positive, neutral, and negative. The experimental analysis states that, the AdaRNN attains better accuracy. [11] projected Hierarchical Long Short-Term Memory (HLSTM) and accomplished lower accuracy on Weibo tweet data. [12] deployed a new variant of RNN scheme, the Gated RNN (GRNN) that reaches the gradual accuracy. Hence, the studies consider more than 3 sentiment labels.

At the same time, [13] applied LSTM in binary classification of sentiment and achieved minimum accuracy in a movie review [14]. [15] obtained a simulation outcome with higher accuracy under the application of 7 diverse data types by CNN approach with a convolutional layer. [16] deployed 3-way classification and reached reasonable accuracy with 4 datasets, where optimal three-way model in named as NB-SVM. [17] applied a pre-trained Word2Vec for CNN approach and reaches better accuracies using SemEval-2015 data.

This paper presents an effective DL based Long Short Term Memory (LSTM) model for emotions with intensity level sentiment classification called LSTM-EISC for Twitter text. Initially, the LSTM-EISC model performs preprocessing to remove the unwanted data like hash tags, special symbols, and multiple spaces. A tokenization process also takes place in prior to classification. Then, LSTM model is applied to classify multiple intensities of Twitter data such as Fear, Anger, Joy and Sadness. For experimentation, the results are investigated using SEMEVAL2018 Task-1Emotion Intensity Ordinal Classification dataset.

2. The Proposed Method

The overall process involved in the proposed model is depicted in Fig. 1. At the earlier level, pre-processing takes place to remove the unwanted content and makes it suitable for classification process. Followed by, LSTM model is applied to classify the tweets into different emotions and sentiments in an effective way.

2.1. Problem Formulation

Suppose $X = \{x_{1,} x_2 \ldots x_n\}$ be the collection of instances and $Y = \{y_{1,} y_{2,} \ldots , y_m\}$ be the group of labels. The set of data is expressed as:

□ = □▯, Y_i □▯ ∈ □□□□Y_i ⊆
Y▯□▯*h* □□□□□□□▯□□□□▯□□□□□□▯□□□□□□□□▯▯*h* □▯ (1)

Here, D implies a supervised multi-label dataset. The process of multi-label classification is very complex due to the development of a label in an exponential manner as values of class labels get improved. The general criterion has been presented to report these problems, in which the problems are converted into a conventional classification issues. The main aim is to characterize the learning operation by applying the label associations. According to the list of correlations, the classical transformation models are combined into 3 methods like first-order, second-order, as well as high-order approaches.

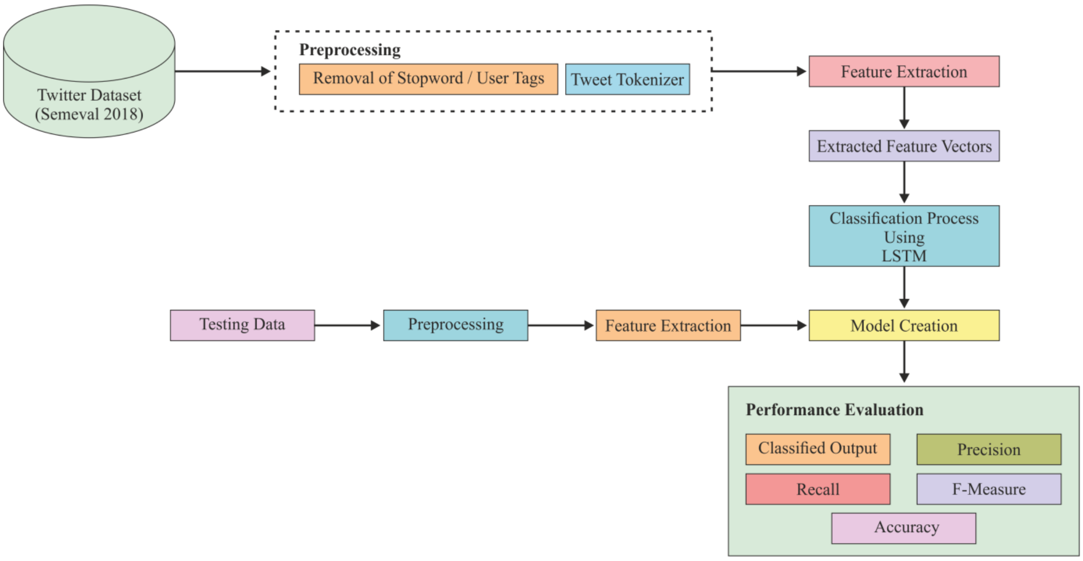

Figure 1. Block diagram of LSTM-EISC model

Initially, first-order approaches degrade the issues into few autonomous binary classifying problems. At this point, 1 binary classifier has been learned for every feasible class, and removing the co-existence of alternate labels. Hence, the count of autonomous binary classifiers should be same as number of labels. For every multi-label training instance $(x_{i,}Y_{i})\in D_{,}y_k \in Y$, it is developed with a binary classification training set, D_k as given in the following: x_i might be referred as a positive instance when $y_k \in Y_i$ and negative example. Firstly, the training example may be in the form of $(x_{i,}1) \in D_k$,which results in $(x_{i,}0) \in D_k$ in the alternate case.

Therefore, for every labels $\{y_{1,}y_{2,}\dots{}_{,}y_m\} \in Y_{,}$m training sets $\{D_{1,}D_{2,}\dots{}_{,}D_m\}$ were deployed. According to every training set D_k , binary classifier are trained under the application of familiar learning modules like AdaBoost, -nearest neighbor k(kNN), decision trees (DT), random forests (RF) and so on. The major benefit of first-order methods is the conceptual simplicity as well as maximum efficiency. Therefore, the techniques are less effective because of the removal of label correlations.

Second-order approaches attempts to report the absence of developing label correlations under the application of pair-wise associations among the labels. The pair-wise relationships are used in training 1 binary classifier for all pair of labels. Though a second-order approach performs quite-well in various applications, it is very tedious when compared with first-order approaches with respect to number of classifications. The main difficulty is quadratic, as required classifiers are $\begin{matrix}m\\2\end{matrix}$. Furthermore, in real-time domains, label correlations are said to be highly tedious. High-order approaches are capable of managing the multi-label learning issues by searching high-order associations between the labels. It is achieved by considering linear combinations, a nonlinear mapping, across the entire label space. Even though high-order methods are robust correlation-modelling abilities when compared with first-order as well as second-order counterparts, which are computationally requirement and minimum reliable.

2.2. Pre-processing

To convert raw tweets to be adaptable for classification, a series of preprocessing steps takes place. Initially, the emojis present in the tweet are converted into the respective Unicode followed by transformation into lexicon. For instance, the emoji □ is transferred to the corresponding Unicode of U+1F628 and is again transformed into the lexicon of “Fearful”. Then, tags and hash tags are also removed followed by stop words removal. Since the stop words do not signify any meaning in the twee, it is discarded from the tweet. In the same way, the special symbols and numerals are also deleted to avoid confusion in the vector generation process. Afterwards, Tokenization process is carried out by the use of Tweet Tokenizer, a Twitter-aware technique from NLTK.

2.3. LSTM Architecture

LSTM learns dependencies that range over random longer time intervals. LSTM resolve the diminishing gradients issue by interchanging a normal neuron by a difficult structure named as LSTM unit. An LSTM unit is developed using simpler nodes that is linked in a particular fashion. The major units of LSTM structure [18] is shown in Fig. 2 and also given in the following:

1. Constant error carousel (CEC): A core unit with a recurrent link along with a unit weight. The recurrent connection shows a feedback loop at time step as same as 1. The CEC’s activation is inner state that is treated as memory for previous data.

2. Input Gate: A multiplicative unit that secures the data saved in CEC from the interruption of irregular inputs.

3. Output Gate: A multiplicative unit that secures alternate units from the interference by data recorded in CEC.

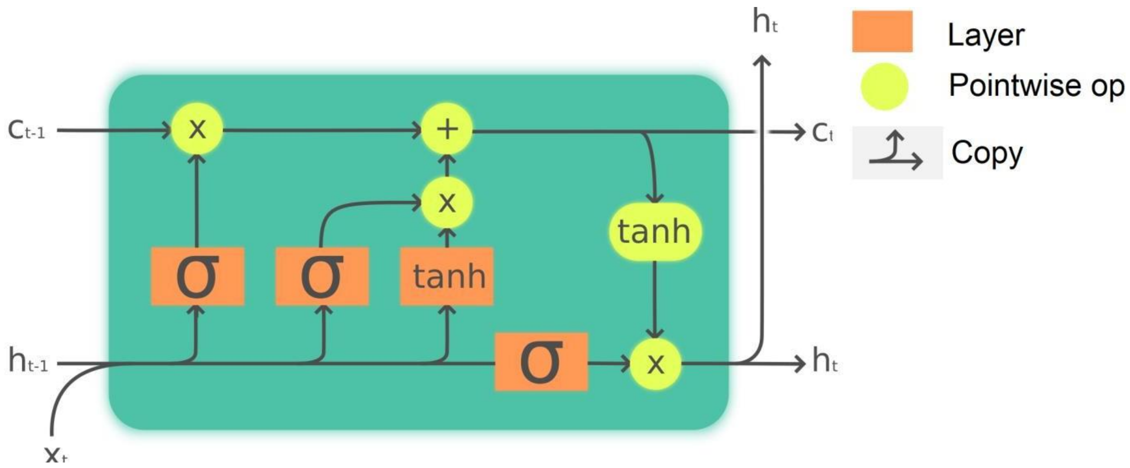

Figure 2. General Architecture of LSTM

The input as well as output gate control enables CEC. At the time of training phase, the input gate trains the data within CEC. In line with this, input gate is assigned with a value of 0. Likewise, the output gate learns the time of letting the data flow from a CEC. If these gates are closed, activation is terminated within a memory cell. It enables the error signals to flow over the issue of diminishing gradients.

The structures of an LSTM unit with forget gate and it is applied for remaining issues. The major units of LSTM unit are given in the following:

1. Input: The LSTM unit consumes recent input vector represented by x_t and output derived from existing time step shown by h_{t-1}. The weighted inputs are consolidated as well as conveyed by tanh activation that results in z_t.

2. Input gate: This metric reads x_t and h_{t-1}, determines the weighted sum, and utilizes sigmoid activation. Hence, the simulation outcome is improved with the z_t, to offer input flow into memory cell.

3. Forget gate: This gate is a process from where the LSTM trains to reset the memory data while it becomes old and non-related. This happens when a network computes the novel sequence. The forget gate reads x_t and h_{t-1} and implements a sigmoid activation to weighted inputs. The result, ft is increased by a cell state at existing time step i.e. s_{t-1} that enables to forget the memory data that is non-essential.

4. Memory cell: This is composed with CEC, and recurrent edge with unit weight. The recent cell state s_t is determined by removing irregular data from previous time step and approving related data from the current input.

5. Output gate: It is employed with the weighted sum of x_t and h_{t-1} and utilizes sigmoid activation to manage the data flow from LSTM block.

6. Output: The output or result of a LSTM unit, $h_{t,}$ is estimated by conveying a cell state c_t by a tanh and improving with output gate, o_t. The performance of a LSTM is presented by the given equations:

$$i_t = \sigma(W_i \cdot \quad [w_{t,} h_{t-1}] + b_i) \tag{2}$$

$$f_t = \sigma \;\; W_f \cdot \quad w_{t,} h_{t-1} \quad + b_f \tag{3}$$

$$o_t = \sigma \;\; W_0 \cdot \quad w_{t,} h_{t-1} \quad + b_0 \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh \;\; W_c \cdot \quad w_{t,} h_{t-1} \quad + b_c \tag{5}$$

$$h_t = 0_t \odot \tanh \;\; c_t \tag{6}$$

where i, f and 0 denotes input, forget and output gates correspondingly. Here, σ indicates the sigmoid function utilized for controlling in and out details at every round. Besides, $\{W_i, W_f, W_o, W_c, b_i, b_i, b_o, b_c\}$ are theparameters to be learned during training.

2.4. LSTM for EISC Process

Mostly, the existing works are comprised with major advantages of target word presentations that fail to show the aspects in an accurate manner and tend to generate negative aspect as specific presentations through the model for lat sentiment classification operation. Besides, aspect embeddings are named as modelling parameters which has been trained at the time of training. In this approach, LSTM applies context word vector as well as aspect embedding as input. Hence, the aspect embeddings are applied to help the major approach in finding massive data relevant to the concerned aspect.

The equations determining $i_t, f_t, 0_t, c_t$ are interchanged by the following:

$$i_t = \sigma\ W_i \cdot \quad w_t, v_a, h_{t-1} \quad + b_i \tag{7}$$

$$f_t = \sigma(W_f \cdot \quad [w_t, v_a, h_{t-1}] + b_f) \tag{8}$$

$$o_t = \sigma(W_o \cdot \quad [w_t, v_a, h_{t-1}] + b_o) \tag{9}$$

c_t

$$= f_t \odot c_{t-1} + i_t \odot \tanh\ W_c \cdot \quad w_t, v_a, h_{t-1} \quad + b_c \tag{10}$$

where $v_a \in R^{d_a}$ is the aspect embeddings matrix M. $\Theta^{lstm} = \{W_i \in R^{d\times(2d+d_a)}, W_f \in R^{d\times(2d+d_a)}, W_0 \in R^{d\times(2d+d_a)}, W_c \in R^{d\times(2d+d_a)}, b_i \in R^d, b_f \in R^d, b_0 \in R^d, b_0 R^d\}$ are implied as model parameters.

2.3.1. Stage I: Position Attention

Initially, it establishes the position based on intuition which has a polarity that the provided aspect is selected to a higher extent by neighboring context words and affected by the far apart context words. It assures the ordinary rules of criticism in natural language. The Stage I method is developed to report the aspects present in a sentence in a sequential manner. The Gaussian kernel has been employed in estimating the position-aware influence propagation:

$$\square\square\square\square\square\square(\mu) = \exp\left(\frac{-\mu^2}{2\gamma^2}\right) \tag{11}$$

where μ is a distance among recent context word and predefined aspect, and γ implies a propagation value. It is considered that the scope of particular distance follows the Gaussian distribution across every dimension. Hence, the influence is expanded into a matrix named as P. The influence in i-th dimension as the distance of μ can be evaluated as:

$$\square(\square, \square) \sim \square\ (\square\square\square\square\square\square(\mu), \sigma') \tag{12}$$

where $P(i, \mu)$ applies a normal distribution with mean rate of Kernel (μ) and standard deviation (SD) of σ. It is pointed that, every columns of P implies an influence vector. For simplicity in function, is noted as $p_j \in R^{d_p}$ from P as an influence vector for j-th context word with a distance of , and μ d_p indicates the dimension of position vector. The production of position-aware influence vectors.

Particularly, to acquire the i-th aspect term of review, by providing the aspect words, the attention distribution of context word at position j in a sentence might be expressed as:

$$A^i_j = \frac{\exp(e(h_j, v^i_a, p_j))}{\sum_{k=1}^{n} \exp(e(h_k, v^i_a, p_k))} \qquad (13)$$

$$e(h_j, v^i_a, p_j) = \eta^T \tanh(W_h h_j + W_p p_j + W_a v^i_a + b) \qquad (14)$$

where $e(h_j, v^i_a, p_j)$ denotes a value function which estimates the semantic among the j-th context word as well as aspect. $\Theta^{att} = \{W_h \in R^{d\times d}, W_p \in R^{d\times dp}, W_a \in R^{d\times da}, b \in R^d, \eta \in R^d\}$ are described as training parameters. Thus, the aspect-based sentence presentation is attained:

$$r_i = \sum_{j=1}^{n} A^i_j h_j \qquad (15)$$

Consequently, the aspect-based attentive presentation r undergoes into target space of C classes:

$$r = \tanh(W_r r_i + b_r) \qquad (16)$$

where $\Theta^{(classifier)} = \{W_r \in R^{d\times C}, b_r \in R^C\}$ are the modelling parameters. Then, a softmax layer has been applied to determine the sentiment distribution:

$$g_c = \frac{\exp(r^c_i)}{\sum_{z=1}^{C} \exp(r^z_i)} \qquad (17)$$

2.3.2. Stage II: Modeling Process

Assuming the disturbance from several aspects, at this point, it attempts to develop multi-aspects simultaneously. By deriving, it is applied with Frobenius for creating penalization term that is same as L_2 regularization norm. $A = \{A_1, A_2, \cdots, A_m\}$ represents a matrix deployed by attention weights of aspects in sentence, and $A_i = \{A^1, A^2, \cdots, A^n\}$ refers the attention probability distribution for i-th aspect in a sentence determined by the position attention. To develop an effective aspect that aims on diverse portions of a sentence and limit the disturbance between the aspects, the penalization norm applied is represented as:

$$P = \| AA^T - I \|^2_F \qquad (18)$$

where F is a Frobenius norm and I implies an identity matrix. At this point, it employs fine tune the pre-trained method at primary stage with a penalization term. Finally, i-th aspect-based attentive presentation is determined as:

$$R_i = \sum_{j=1}^{n} A^i_j h_{ij} \qquad (19)$$

where A^i_j is estimated as same as α_j. Later, the aspect-specific attentive representations are considered as input for the classification model.

2.3.3. Model Training

In this implementation, every parameters used in this model are termed as $\Theta = \{\Theta^{lstm}, \Theta^{att}, \Theta^{(classifier)}, M\}$. It is evident that shows M the aspect embeddings. Here, it is selected with cross entropy with L_2 regularization as a loss function for optimizing the method:

$$L = -\sum_{e\in E}\sum_{c=1}^{c} y_c\ e \cdot \log\ g_c\ e\ + \lambda L_2\ \Theta \qquad (20)$$

where E represents a dataset, e signifies an individual sample, y_c (e) implies the optimal sentiment distribution and λ showcases the coefficient for L_2 regularization. It has been noted that, first stage model as well as second stage shares the network structure in general; however the penalization term under the application of Frobenius norm has been applied to fine-tune the second stage approach. The entire process involved in the LSTM based sentiment classification is shown in Fig. 3. When the strength exceeds the threshold value, anyone of the Emotion (i.e. Anger, Fear, Joy, Sadness) will be generated. The strength value obtained can be mapped onto one of different Emotion Classes.

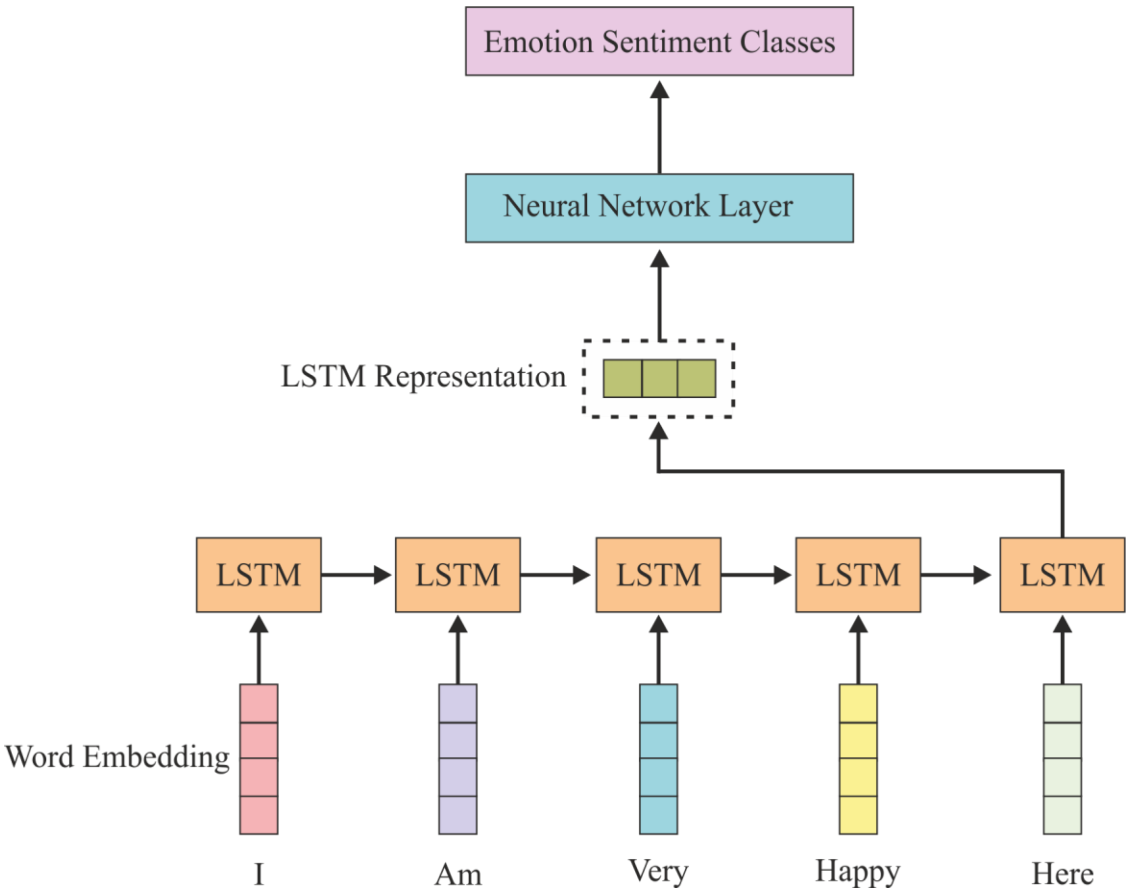

Figure 3. Process involved in LSTM based Sentiment classification

3. Performance Validation

In order to ensure the efficient function of a proposed system, a series of experiments were carried out with the application of Python Programming language. The utilized dataset and the obtained results are analyzed under several subsections.

3.1. Dataset

To validate the result of proposed LSTM-EISC model, a standard SEMEVAL2018 Task-1Emotion Intensity Ordinal Classification dataset is applied [19]. It is composed with a set of 4042 tweets with 4 intestines such as joy, fear, anger, and sadness. From the provided tweets, 1074 tweets belong to Joy, 650 tweets come under the class of fear, 991 tweets come under the anger while 555 tweets exist from the class of sadness. The dataset is divided into training as well as testing data in the ratio of 7.5:2.5. The data related to a dataset has been listed in Table 1.

Table 1 Dataset Description

Description	Values
Total Number of Affect Dimension	3270
Number of Instances in Anger (Labelled: 0)	991
Number of Instances in Fear (Labelled: 1)	650
Number of Instances in Joy (Labelled: 2)	1074
Number of Instances in Sadness (Labelled: 3)	555

3.2. Results analysis

Table 2 has offered the obtained classification results of affect dimension using LSTM-EISC model under several measures. The table values indicated that the instances under anger class are effectively classified with the precision of 93.07%, recall of 89.50%, F-measure of 91.26% and accuracy of 94.70% respectively. Simultaneously, the instances under fear class are effectively classified with the precision of 90.63%, recall of 87.85%, F-measure of 89.22% and accuracy of 95.66% correspondingly. On the same way, the instances under joy class are successfully classified with the precision of 97.22%, recall of 97.95%, F-measure of 97.59% and accuracy of 98.31%. Similarly, the samples under sadness class are effectively classified with the precision of 87.44%, recall of 95.31%, F-measure of 91.20% and accuracy of 96.75% correspondingly. On average, the proposed method has attained a maximum precision of 92.09%, recall of 92.65%, F-measure of 92.32% and accuracy of 96.36%.

Table 2 Performance Measures of Affect Dimension using Proposed LSTM-EISC

Measures	Precision	Recall	F-Measure	Accuracy
Anger	93.07	89.50	91.26	94.70
Fear	90.63	87.85	89.22	95.66
Joy	97.22	97.95	97.59	98.31
Sadness	87.44	95.31	91.20	96.75
Average	**92.09**	**92.65**	**92.32**	**96.36**

Fig. 4 depicted the accuracy graph derived at the time of training and validation process of the model creation under varying number of epochs. The figure clearly pointed that the validation accuracy is slightly better when compared with training accuracy. On the other hand, it is monitored that accuracy gets improved with increased epochs.

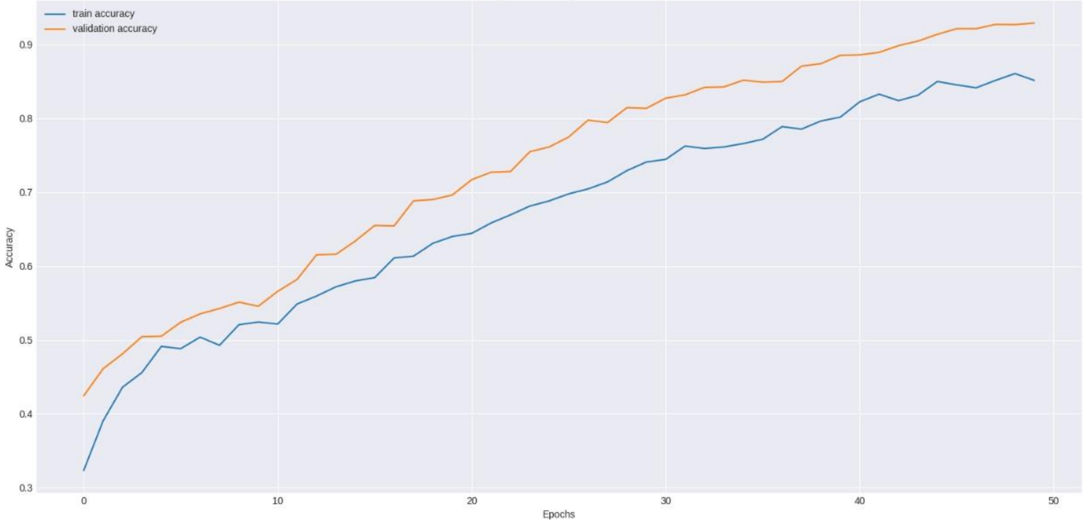

Figure 4. Accuracy Analysis of Training and Validation at the time of Model Creation

Fig. 5 showcases the loss graph accomplished during the training as well as validation process of the model development under diverse number of epochs. From the figure, it is clear that validation loss is moderately better than the training loss. Besides, it is monitored that loss gets enhanced with a rise in number of epochs.

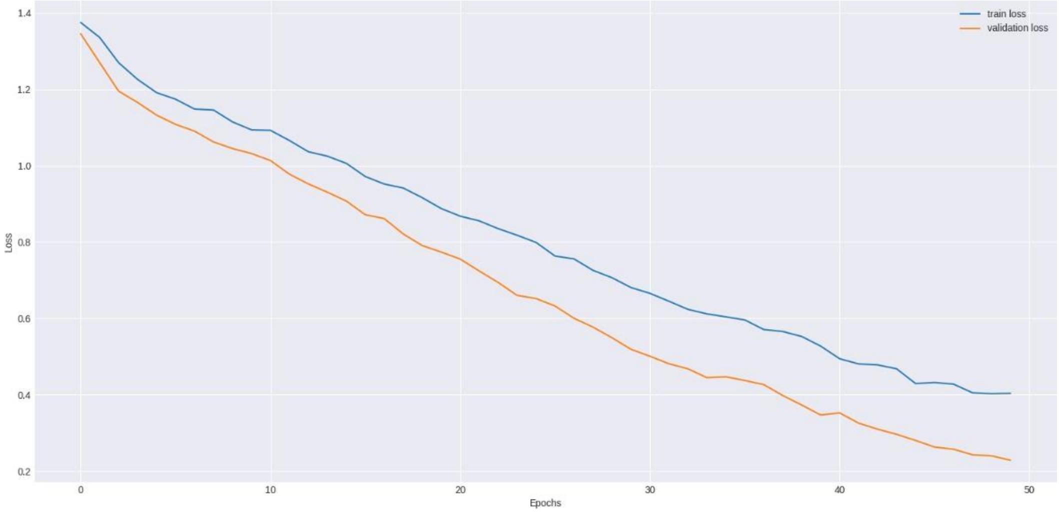

Figure 5. Loss Graph of Training and Validation at the time of Model Creation

The analysis of the results generated by the LSTM-EISC model is examined under various intensities as depicted in Table 3 and Figs. 6-7. It is shown that, the dimensions are classified with average precision of 92.09%, recall of 92.65%, F-measure of 92.32% and accuracy of 96.36%. Concurrently, the LSTM-EISC model has classified the classes under anger intensity with the average precision of 76.83%, recall of 74.41%, F-measure of 75.25% and accuracy of 89.94% respectively. Simultaneously, the LSTM-EISC model has classified the classes under fear intensity with the average precision of 77.66%, recall of 73.06%, F-measure of 73.32% and accuracy of 92.98%.

In line with this, the LSTM-EISC model undergoes classification under joy intensity with the average precision value of 77.07%, recall of 71.96%, F-measure of 73.32% and accuracy of 92.98%. On the same way, the LSTM-EISC model has classified the classes under sadness intensity with maximum precision of 92.20%, recall of 89.87%, F-measure of 90.80% and accuracy of 95.49% correspondingly. As whole, the LSTM-EISC model has classified the applied dataset with average precision of 83.17%, recall of 80.39%, F-measure of 81.03% and accuracy of 92.04% respectively.

Table 3 Result Analysis of Affect and Intensity in terms of Average Precision/Recall/F-Measure/Accuracy

Measures	**Precision**	**Recall**	**F-Measure**	**Accuracy**
Affect Dimension	92.09	92.65	92.32	96.36
Intensity-Anger	76.83	74.41	75.25	89.94
Intensity-Fear	77.66	73.06	73.32	92.98
Intensity-Joy	77.07	71.96	73.47	85.42
Intensity-Sadness	92.20	89.87	90.80	95.49
Average	**83.17**	**80.39**	**81.03**	**92.04**

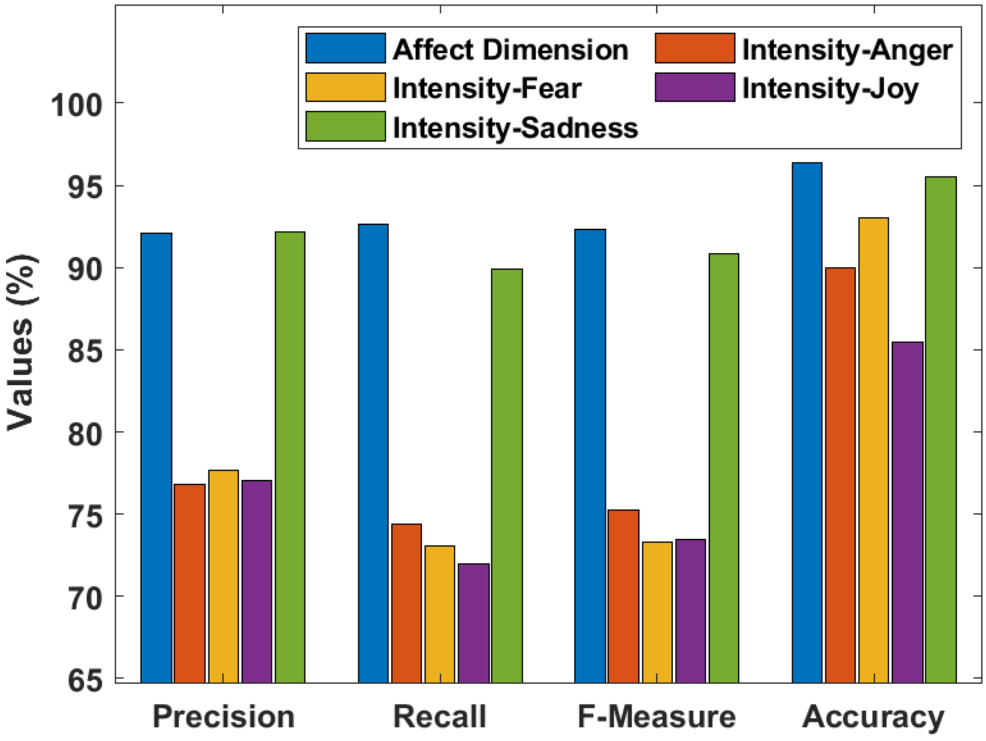

Figure 6. Classifier result analysis of LSTM-EISC model

ISSN: 2233-7857 IJFGCN

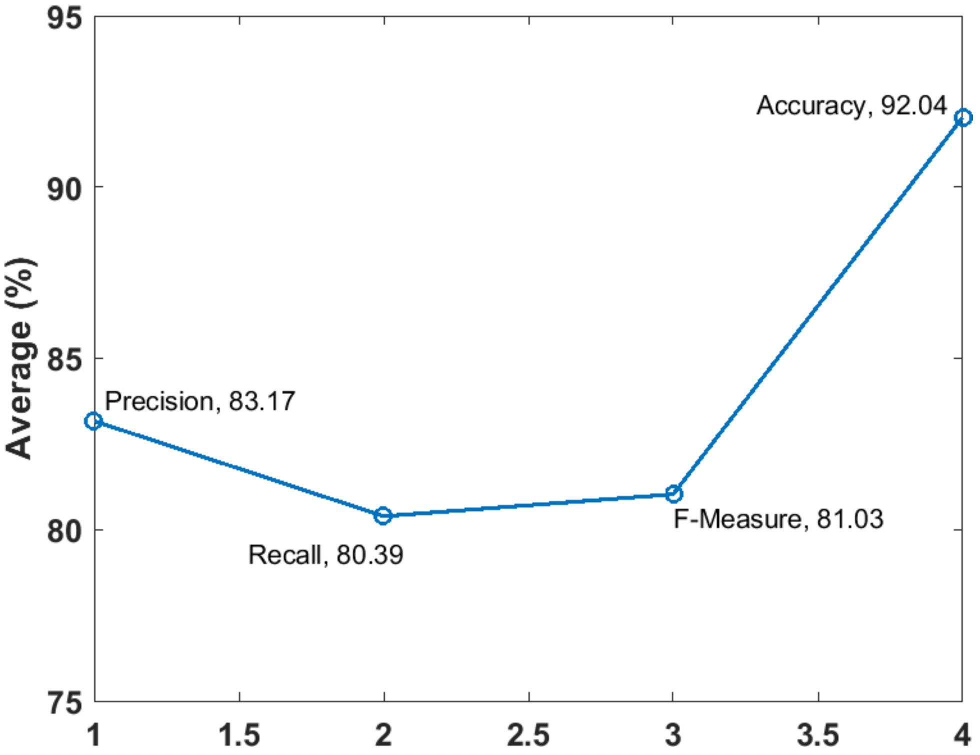

Figure 7. Average result analysis of LSTM-EISC model

Table 4 Result analysis of Existing with Proposed LSTM in terms of Accuracy

Methods	Dataset	Accuracy
LSTM-EISC	**SemEval-2018**	**92.04**
CNN	SemEval-2018	82.72
Multiple Logistic Regression (MLR)	SemEval-2018	62.27
MNB	SemEval-2018	54.70
Random Forest	SemEval-2018	54.00
Linear SVC	SemEval-2018	48.90
GRU Model	SemEval-2018	52.40
Context Aware Model	SemEval-2018	53.20
MohameedJabreel et al.,	SemEval-2018	59.00
Mondher Bouazizi et al.,	Tweets	60.20
Malak Abdullah et al.,	SemEval-2018	59.90

Table 4 and Fig. 8 have demonstrated the final result of comparative analysis of the LSTM-EISC approach with the currently proposed methods by means of accuracy. From the figure, it is evident that the linear SVC model has offered poor result with the minimum accuracy of 48.90%. On the other hand, it is apparent that GRU and Context Aware models has provides impractical outcome with the least accuracy measures of 52.40% and 53.20% correspondingly. In addition, the MNB and RF approach has shown a slightly higher and near identical results by providing accuracy values of 54.70% and 54% respectively. Likewise, the methods projected by Mohameed Jabreel et al., Mondher Bouazizi et al., and Malak Abdullah et al. leads to identical and gradual results with the accuracy of 59%, 60.20% and 59.90%. Furthermore, the MLR and CNN model has performed quit-well than other models by resulting higher accuracy of 62.27% and 82.72%. Therefore, the proposed LSTM-EISC model has attained to best classification outcome and accomplished a higher accuracy measure of 92.04%.

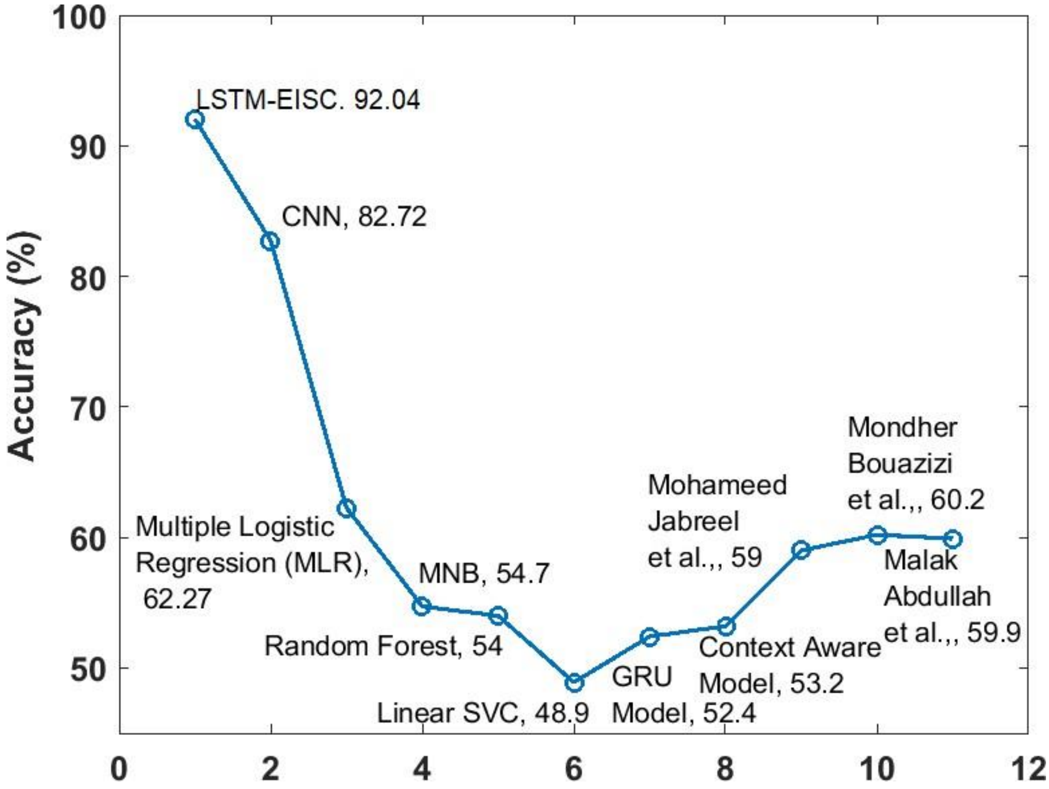

Figure 8. Comparative accuracy analysis of LSTM-EISC with existing models

From the above-mentioned extensive experimental analysis, it is evident that the proposed LSTM-EISC model has outperformed all the existing methods in a significant manner. Therefore, it can be considered as an appropriate tool for the emotion classification of content in social networking sites like Twitter, Facebook and so on.

4. Conclusion

This paper has introduced an effective LSTM-EISC model for emotions with intensity level sentiment classification called LSTM-EISC for Twitter text. At the earlier level, pre--processing takes place to remove the unwanted content and makes it suitable for classification process. Followed by, LSTM model is applied to classify the tweets into different emotions and sentiments in an effective way. For experimentation, the results are investigated using SEMEVAL2018 Task-1Emotion Intensity Ordinal Classification dataset. The detailed comparative analysis ensured the effective classification performance of the LSTM-EISC model with the maximum average precision of 83.17%, recall of 80.39%, F-measure of 81.03% and accuracy of 92.04%. From the detailed experimentation, it can be inferred that the LSTM-EISC model can be implemented in real time to analyze the Twitter data for identifying people's opinion in real time situations.

References

[1] B. Liu, “Sentiment analysis: Mining opinions, sentiments, and emotions”, Cam- bridge University Press, (2015).

[2] B. Liu, "Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies", Morgan & Claypool Publishers, (2012).

[3] https://hackernoon.com/text-processing-and-sentiment-analysis-of-twitter-data-22ff5e51e14c.

[4] M. Ganapathibhotlaand Liu, B, "Mining opinions in comparative sentences",Proceedings of the 22nd international conference on computational linguistics (COLING), (2008)241–248.

[5] Y. Liu, X. Yu, Z. Chen and B. Liu, "Sentiment analysis of sentences with modalities",Proceedings of the 2013 international workshop on mining unstructured big data using natural language processing, (2013)39–44.

[6] X. Ouyang, P. Zhou, C.H. Li and L. Liu, "Sentiment analysis using convolutional neural network", In Proceedings of the 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, (2015) 2359–2364.

[7] R. Socher, B. Huval, C.D. Manning andA.Y.Ng, "Semantic compositionality through recursive matrix- vector spaces", Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, (2012) July 1201–1211.

[8] J. Deriu, A. Lucchi, V. De Luca, A.Severyn, S. Müller, M.Cieliebak, T. Hofmann and M. Jaggi, "Leveraging large amounts of weakly supervised data for multi-language sentiment classification", Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, April (2017) 1045–1052.

[9] I. Chaturvedi, E. Cambria, R.EWelsch and F. Herrera, " Distinguishing between facts and opinions for sentiment analysis: Survey and challenges", Inf. Fusion,(2018), pp. 65–77.

[10] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou and K. Xu, "Adaptive recursive neural network for target- dependent twitter sentiment classification", Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Baltimore, Maryland, USA, June (2014) 49– 54.

[11] M. Huang, Y. Cao and C. Dong, "Modeling rich contexts for sentiment classification with lstm",arXiv 2016, arXiv:1605.01478.

[12] D. TangB. Qin and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification", Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, September (2015) 1422–1432.

[13] Q. Qian, M. Huang, J. Lei and X. Zhu, "Linguistically regularized lstms for sentiment classification",Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Canada, July (2017)1679–1689.

[14] B. Pang, L. Lee,"A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts", Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, July (2004) 271–278.

[15] Y. Kim, "Convolutional neural networks for sentence classification",Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Qatar, October (2014) 1746–1751.

[16] Y. Zhang, Z. Zhang, D. Miao, andJ. Wang, " Three-way enhanced convolutional neural networks for sentence-level sentiment classification" Information Science, (2019), pp. 55–64.

[17] A. Severyn andA.Moschitti, "Unitn: Training deep convolutional neural network for twitter sentiment classification", Proceedings of the 9th International Workshop On Semantic Evaluation (SemEval 2015), Denver, Colorado, June (2015) 464–469.

[18] https://en.wikipedia.org/wiki/Long_short-term_memory

[19] https://competitions.codalab.org/competitions/17751