Volume 10, Issue 5
October 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Tweet Sentiment Analysis using Deep Learning Technique

Md. Ahsan Habib

Department of Computer Science & Engineering, Mawlana Bhashani Science and Technology University , Dhaka Bangladesh

E-mail:habib5972@gmail.com

ABSTRACT

Recently classification research has become the most common and crucial aspect in Natural Language Processing(NLP).In machine learning approach various types of classification such as multi-level, multi-class and binary classification are discovered. Sentiment analysis is defined as the process of identifying sentence polarity like positive or negative. Sentiment analysis is the field of study that analyzes emotions, attitudes and reviews from different public opinions. Nowadays people express their opinion vastly through social media like Facebook, Twitter, YouTube and different E-commerce sites. To be a successful businessman or politician human opinion is most important. For any kind of decision as a company holder or stockholder sentiment analysis is necessary. In this paper, a hybrid deep learning model has been proposed that can detect positive or negative sentence by analyzing the user's tweet. Deep learning algorithms were trained using training data and then output has been evaluated on the test data of the existing Kaggle dataset that was previously released for sentiment analysis. In specific, a proposal was made for a Long Short Term Memory(LSTM) with various word embedding strategies which yielded positive results relative to Convolutional Neural Network(CNN).

Keywords: *Natural Language Processing(NLP), Sentiment, Sentiment Extraction, Tweet, Deep Learning, Long Short Term Memory(LSTM), Word Embedding.*

1. INTRODUCTION

The speedily growing field of web data analytics has started to play a pivotal role in the advancement of business and political polarities. People try to find reviews on reputation and popularity at a specific moment of a different brand. It is difficult to find these opinions manually. There are some of the obstacles posed by the opinion or sentiment of the client or employer:

- Textual Viewpoint (Posts/ Tweet) with related textual content).
- Geo-spatial viewpoint (tweet places nearby).
- Temporal View (Invariant Time intervals dependent on the timeline of the user).
- For manufacturers, sentiment analysis (SA) can act as a method to evaluate their goods and services on the basis of digital payment and e-commerce channels on Twitter.

There have several ways to analyze sentiment automatically. Deep learning is a growing field of automation in a different sector. For this reason, we are prompted to use deep learning in finding sentiment analysis with better accuracy. Finding out about the thought of other people is an important part of gathering information. According to [1], the goal of Natural Language Processing is described as: "To accomplish humanlike processing of natural language". In [2] describes that due to resources of review such as online sites and different types of blogs, arise new challenges when if we try to find out and understand the opinions of people.

A system was proposed by Pang and Lee [3] where an opinion of positive or negative was found out by the ration between the positive words to the total words. In [4] states that when analyzing a text, the context of the sentence can give the actual meaning rather than the word in isolation. Jiang et al. [5] focus on target-dependent sentiment classification. Here target dependent means whether the sentiment is positive, negative, or neutral depends on the nature of the question that is asked. To deal with the tweet sentiment analysis problem here we give the key objectives of our paper are as follows:

- First, we introduced a new hybrid method using deep learning methods to find the negative or positive view of the sentence.
- Second, we prepossessed the data with converting html entities, apostrophe lookup and remove stop words to clean the tweets. Then we apply data label encoding along with vectorizing the text data then pad the text in a specific length.
- Third, we apply several feature sets of LSTM and get the accuracy of different features.
- Finally, we use a sigmoid activation function in hidden layer 03 to convert the fully connected layer output and able to detect the positive or negative sentence.

The rest of the paper is structured as follows: Section 2 discusses similar work on sentiment analysis and modern techniques. The methodology proposed for Section 3 accounts. Experimental findings and analysis are discussed in section 4. This article is eventually concluded in the last section 5 and Section 6 we have discussed our future work.

2. BACKGROUND

Recently young generation has involved with online activities like Facebook, Twitter, YouTube and E-commerce site which come with a large number of public reviews in the form of text like comments, posts, status. Sentiment analysis is very crucial task in Natural Language Processing. However, we categorize these works.

A. Literature Review

Recurrent Neural Networks (RNN) can do short-term dependencies in a chain of text [6]. But RNNs cause trouble when it deals with long-term dependencies which have a vast impact on the meaning as well as the comprehensive polarity of a document. This problem can be solved by Long Short Term Memory (LSTM) networks by creating a memory into the network.

Twitter is one of the rich resources for opinions of various events and products. Several researches are going on Twitter dataset. The sentiment detection and analysis of these microblogs is one of the most challenging tasks that has brought an increased research interest in recent years [7]. The paper declared that traditional RNNs are not sufficient to deal with complex sentiment expressions. Therefore they implemented an LSTM network for classifying the sentiment of tweets. Another experiment on sentiment classification on four large datasets presented in[8]. The first three datasets are reviews of the restaurants from Yelp between the years 2013 to 2015. The second dataset contains reviews of the movies from the IMDB, classified as positive or negative reviews. In this paper, an LSTM model is compared with different classifier models e.g. SVM where LSTM performed better.

Richard et al. [9] proposed the Recursive Neural Tensor Network (RNTN) architecture that presents a phrase using word vectors that include a parse tree. Additionally, it has distinct multiplicative units to regulate the information flow of the network. The memory block carried an input gate and an output gate [10]. Mohamed et al. [11] illustrated that deep LSTM RNNs is also used successfully for speech recognition. For speech recognition and language modeling, LSTM presented a remarkable accuracy. Deep LSTM RNNs have also been used successfully for speech recognition.

The depth in deep LSTM RNNs means the input to the network at a given time step goes through multiple layers of LSTM in addition to the propagation through time and layers of LSTM. It is used to eliminate the overfitting problem of the network. It is computationally more expensive but offers a very reduced model for the task [12]. Zhou et al. [13] proposed an LSTM network based on attention in order to learn whether the document contains the reviews in English and Chinese by traversing word vectors as text representation. RNN is often applied for handwriting and speech recognition. An LSTM unit is proficient in remembering values for a short or a long period [8] without any activation function within its recurrent components. It also solved the vanishing gradient problem as the value stored is not iteratively flattened over time. Deep neural networks (DNNs) used in different NLP tasks like modeling the language [14] have gained a significant performance. Sentiment analysis is also a dynamic field of research [13]; it also uses to identify the opinions from a text.

For research purposes, text collected from different sources such as the review of the user. The users of microblogs express their views towards a particular event or product. Due to the contextual limit, sentiment analysis from the microblog is a challenging task. Collobert et al. [15] presented a model that deals with sentiment analysis based on a supervised learning method. In this model, each data element is specified as either ’positive’ or ’negative’. Recently Deep neural networks (DNNs) achieved notable performance gains on different NLP tasks.

In paper [16] represent sentiment analysis using LSTM and CNN on IMDB comments. This paper proposed a model where a large number of CNN-LSTM layers are combined for analysis. In recent past many contributors have performed multi-label classification in Bengali sentences[17],[18].We have found one related research paper[19] where authors had performed almost 10,000 Bengali sentence with total of eight labels collected from crime type news articles for multi-label classification. Rane et al. [20] suggested various methods of tweet preprocessing accompanied by the usage of the seven algorithms to evaluate the sentiment of the airline service tweets inside the United States. After CNN is added to text classification, the LSTM network is fed directly by the sequential features [21]. This approach helps LSTM to benefit from higher-order features for long-term dependencies.

B. Deep Learning

Deep learning is an artificial intelligence(AI) function that works like human brain to process, detect objects, recognize speech, translate language and make decisions. Deep learning is the form of machine learning algorithm which can be used to detect fraud or money laundering among other functions. Deep learning has brought an explosion of data in all forms and from every region of the world[22]. This data known simply as big data. Deep is subset of machine learning which uses hierarchical neural networks to analyze data. A large number of nodes are linked together within these neural networks like human brain. Deep learning was first theorized in the 1980s[23],there are two main reasons it has only recently become useful:

- Deep learning needs huge number of labeled data. Example if anyone make driverless car ,it requires millions of images and thousands hour of video.
- Deep learning need high capability of computing power. High performance GPUS have a parallel architecture that is efficient for deep learning [23].

C. Recurrent Neural Network

Recurrent neural network is a special type of artificial neural network(ANN) which requires sequential data or time series of data. Like feedforward and convolutional neural networks and recurrent neural networks utilize training data to learn which are distinguished by their “memory” as they take information from prior inputs to influence the current input and output[24].RNNS can take one or more input vectors and produce one or more output vectors and the output(s) are influenced not just by weights applied on inputs like a regular NN, but also by a ”hidden” state vector representing the context based on prior input(s)/output(s)[25]. Moreover, RNN is the precursor to LSTM. In RNN a single time step of the input is provided to the network. Then it calculate its state using set of current input and the previous state. An RNN remembers each

Volume 10, Issue 5
October 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

and every information through time. It is useful in time series predictions only because of the feature to remember previous inputs as well. This is called Long Short Term Memory.

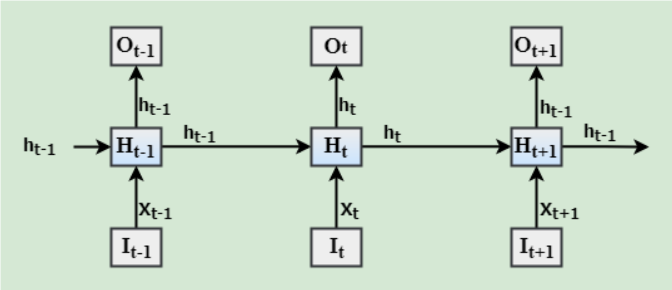

Figure 1: Traditional Recurrent Neural Network.

D. Natural Language Processing(NLP)

Natural Language processing(NLP) refers to the branch of computer science-and more specially ,the branch of artificial intelligence or AI-concerned with giving computers the ability to understand text and spoken words in much the same way human beings can[26]. There are two main phases to natural language processing: data preprocessing and algorithm development.

Data processing means preparing and cleaning text that machine can easily analyze it. There are several ways that can be done:

- Tokenization: This is the way in which text are broken into smaller units.
- Stop word removal: This techniques refers that user remove the unnecessary words and keep the unique words that offer the most information about the text.
- Lemmatization and stemming: This is when words are reduced to their root forms to process
- Parts of speech tagging: This is the way in which words are marked based on their parts of speech.

Once the data has been processed , an algorithm has to be developed to process it. There are two main types of algorithm which are being used.

- Rules based system: In this approach the system has carefully designed using linguistic rules.
- Machine learning based system: Machine learning algorithm use statistical method. They learn to perform tasks based on training data they are fed[26].

3. PROPOSED METHODOLOGY

E. Input data

We use the tweeter analysis dataset to input the algorithm. In our algorithm, we load data directly from the CSV as the dataset is in the CSV format file to train the algorithm.

F. Data Prepossessing

To get a better result, the tweets are being processed. Tweets can have several languages based on the user, so we have to clean the irrelevant data. After that, the URLs and the username are removed. Then we will do case-folding from upper case to lower case letters. Finally, we will remove the stop words as it is not significant and not related to emotions.

a) Create a word to integer: The vocabulary consists of a list of words that occurred in our text document and these words have their index. It will help us to create a vector for a text document. First, we take the sentence then vectorize it and count the number of occurrences in the text. The final vector will be the length of the text and called it featured vector. In this vector, each dimension will numeric or categorical. We use the Count Vectorize provided by a scikit-learn library for vectorizing.

b) Encode the data label: So far, we create a list of review and index mapping dictionary from all our tweeter review. In this step, we replace each word in our document by integers.

c) Encode the label: Our dataset consists of many labels but we use text and sentiment labels in our work. The sentiment has consisted of two output labels: one label is 1 which means positive and another label 0 which means negative.

d) Padding the remaining data: To maintain long or short text in our document we will pad or truncate of our text in a specific length. We call it sequence length which is the same as the number of steps for the LSTM layer.

G. Feature Selection

We interacted with various collections of features. Then a mixture of different set features was used to obtain a better outcome after padding the data. Below is a rundown of the various types of features that we used for our experiment.

a) ***Trainable Embed Features****:* We used the Trainable Embed layer to create the Trainable Embed functionality.

b) ***Word2VecEmbed Features****:* Word2Vec embedding [27] technique has been used to get Word2VecEmbed Features. We used genism [28] library to create Word2Vec model. We trained the model with our train info. After that, we obtained word vectors from the Word2Vec model that was learned. Here, too, every term is represented with a contour of dimension 300.

c) ***Fast text Embed Features****:* Here the embedding of the Fast text technique was used. The method for receiving fast text Embed features is the same as the Word2VecEmbed features.

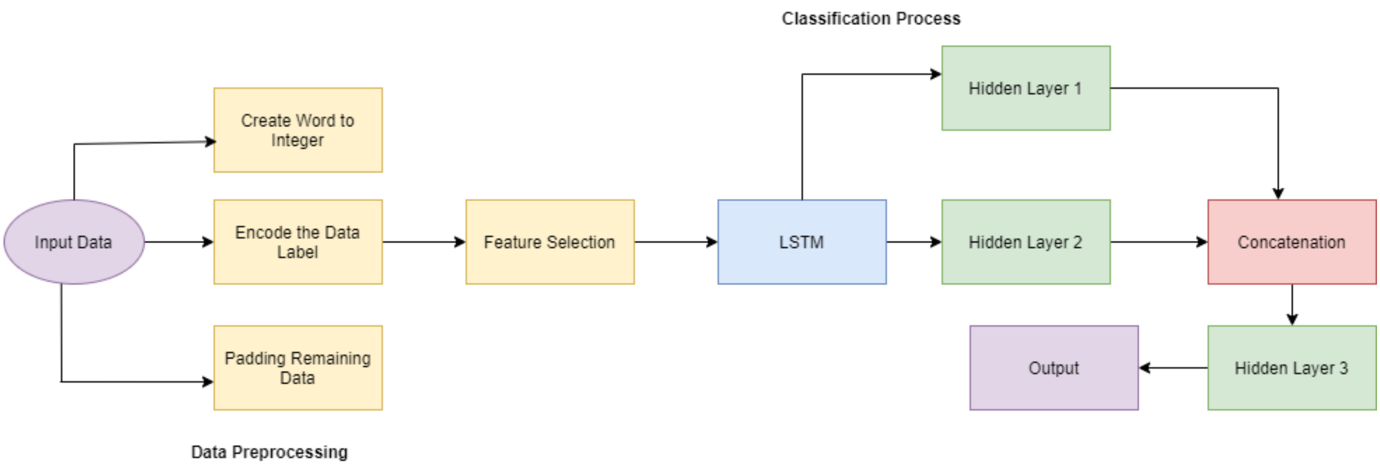

Figure 2: Proposed Model of the System

H. Classification Process

Our proposed model has two parts. They are two parts data preprocessing that has been already described . Now we discuss on another part classification process. Classification process starts with Long Short Term Memory which is one of the special type of recurrent neural network. Then to detect positive and negative sentence embedded functions have been applied to the Long Short Term Memory(LSTM) output layer. Texts are pre-processed using NLP techniques and word embeddings are trained using Word2vec with dimensions 40, 80 and 160, respectively. The hidden layer sizes of the LSTM we used are 40, 80, 120 and 160.The output of the LSTM layer was then fed to the secret layer 1 with an output dimension of 100. Normalized Metadata Features have been fed into hidden layer 2 with a 40-dimension output. After that, the vectors returned to these two hidden layers is concatenated. The output dimension of the concatenation layer was 300. Then a dropout of 0.2 was applied to avoid overfitting. Finally, hidden layer 3 was introduced and the activation function was sigmoid as we did the binary classification for our output. Many of the hidden layers is completely interconnected layers. We set these dimensions by observing the outcomes for a limited amount of test data. The dimensions for which we have seen a better outcome have been preserved. We haven't changed the other settings for the hyperparameter. We've retained them as default values. Figure 1 depicts the proposed methodology

I. Long Short Term Meomry(LSTM)

LSTM is a special type of recurrent neural network. In RNN output of previous step is fed as input for current step. LSTM overcome the long-term dependencies of RNN. It can not predict the word stored in the long term memory but can give more accurate predictions from the recent information. LSTM can by default retain the information for long period of time. LSTM has a chain structure that contains four neural networks and different memory blocks called cells.

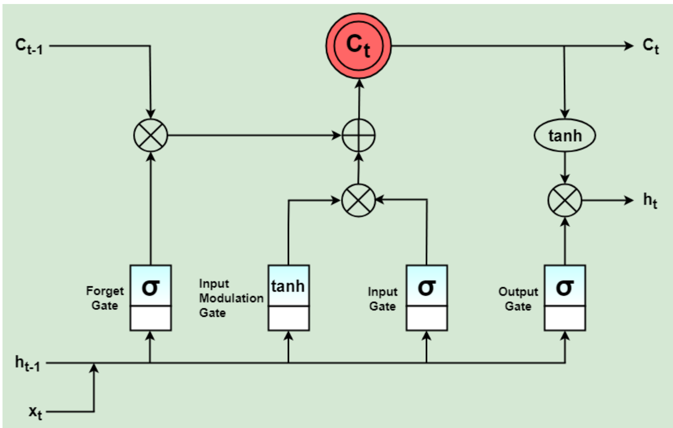

Figure 3: Long Short Term Memory(LSTM)

The LSTM does have the ability to remove or add information to the cell state, carefully regulated by structures called gates. The gates are basically composed out of a activation function layer and a pointwise multiplication operation. There are three basic gates in LSTM. They are:

- Forget Gate: This gate is useful for the information that no longer useful in the cell state. Two input x_t(input at the particular time) and h_t-1(previous cell output) are fed to the gate and multiplied with weight metrics followed by addition of bias. The result is passed through an activation function which gives a binary output. The equation is given below:

Volume 10, Issue 5
October 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

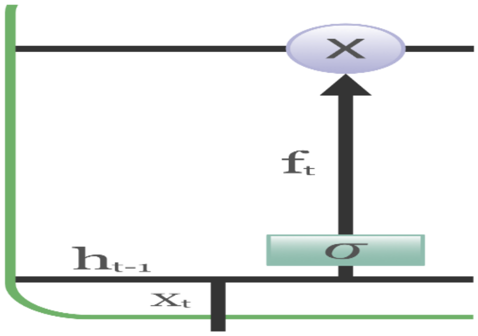

Figure 4: LSTM Forget Gate[29]

- Input Gate: The information which is useful of the cell state is done by input gate. First the information is regulated using the sigmoid and filter the values to be remembered similar to the forgate gate using inputs h_t-1 and x_t. Then a vector is created using tanh function that gives output from -1 to 1.

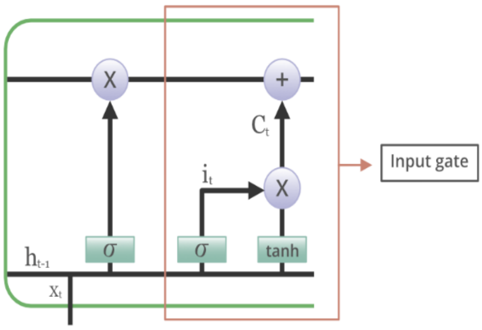

Figure 5: LSTM Input Gate[29]

- Output Gate: The task of extracting useful information from the current cell state to be presented as an output is done by output gate. First, a vector is generated by applying tanh function on the cell. Then, the information is regulated using the sigmoid function and filter the values to be remembered using inputs h_t-1 and x_t. At last, the value of the vector and the regulated values are multiplied to be sent as an output and input to the next cell.

Where W denotes the weight matrices, Ct is the cell state and b is the input bias vector. And the i, f, o are the input, forget and the output gate layer. Cell out activation function in this paper is tanh.

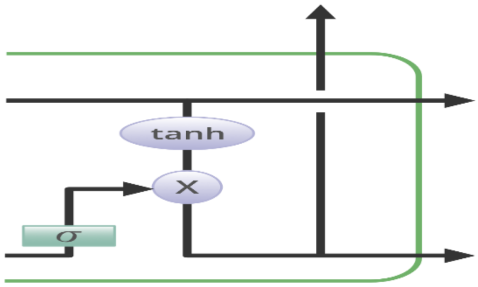

Figure 6 : LSTM Output Gate[29]

An LSTM network computes a mapping from an input sequence x = (x_1, ..., x_T) to an output sequence h = (h_1, ..., h_T) by calculating the network unit activations using the following equations iteratively from t = 1 to T :

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

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

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \quad (5)$$

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

Where W denotes the weight matrices, Ct is the cell state and b is the input bias vector. And the i, f, o are the input, forget and the output gate layer. Cell out activation function in this paper is tanh.

J. Activation Function

Activation function decides whether a neuron should be activated or not by calculating weighted sum and further adding bias with it[30].The key purpose of using this function is to introduce non-linearity into the output of a neuron. There are many types of activation function. In this paper the functions which we used given below:

Sigmoid Function: Sigmoid function is one of the most widely used non linear activation function. This converts the value from 0 to 1.The mathematical expression for sigmoid function is:

```
f(x) = 1/(1+e^-x)
```

Volume 10, Issue 5
October 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

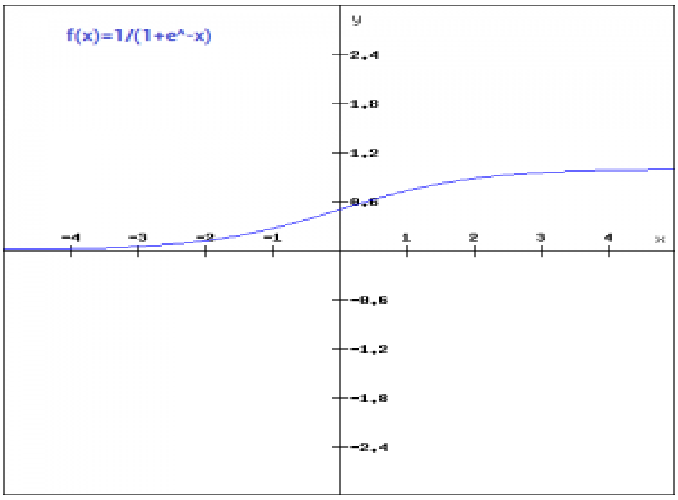

Figure 7: Sigmoid Activation Function[31]

Tanh Activation Function: It is very similar to sigmoid function. But the only main difference between two is Tanh activation function is symmetric. It converts the value from -1 to 1.

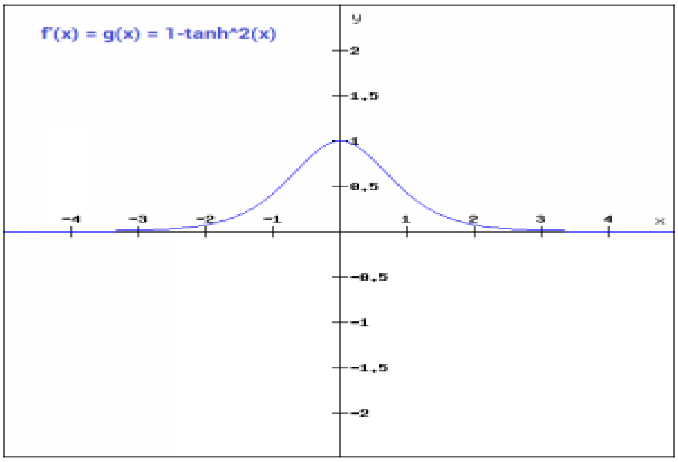

Figure 8 : Tanh Activation Function[31]

4. EXPERIMENT SETUP AND RESULTS

This section provides a brief summary of the datasets used for the evaluation, evaluation metrics, the experiments performed, and corresponding results.

A. Dataset and Experimental Setup

The database has been taken from the Kaggle dataset [32] . It contains 2000 rows and 6 attributes. The sentiment analysis includes only the sentence, value, and sentiment of polarity. The dataset is typically used for different types of analysis from textual data. This research work is aimed at predicting the sentiment from textual data. To attain better efficiency, we tuned some of the hypermeters to train our model using LSTM. We used 90:10, 80:20 and 70:30 ratios for the train test splitting of our dataset.

The attribute is a text data type that represents the sentence of the tweet, a numeric data type that represents polarity which includes either “0” or “1” and a text data type that represents the sentiment polarity shown in Table 1.

Table 1 :An Example of Dataset .

Sentence	Value	Sentiment
I loved to eat.	1	Positive
This product is bad	0	Negative
I miss you very much	0	Negative
I am a good boy	1	Positive
The x party is not elected this year	0	Negative

B. Experiments

The details of three experiments using a different portion of the test and train data set are provided in this section.

Experiment 01: We have experimented with 90% training data and 10% testing data that is selected randomly along with different embedding dimensions hidden dimensions. We have tried all possible scenarios with embedding dimensions 40, 80, 160 and hidden dimensions 40, 80 and 160. In Table II, we set out the highest accuracy obtained for the corresponding ratio and measurements using the LSTM. From Table II, the overall accuracy obtained for LSTM is 84% with a ratio of 70:30 with an embedding dimension 80, hidden dimension 40, where both highest positive accuracy is 59% and negative accuracy is 92% in same ratio.

Table 2:Accuracy Comparison with Trainable Embed Features For LSTM Technique

Features	Train Test Ratio	Positive Accuracy	Negative Accuracy	Total Accuracy
Trainable Embed	90:10	57%	90%	83%
Trainable Embed	80:20	56%	88%	82%
Trainable Embed	70:30	59%	92%	84%

Volume 10, Issue 5
October 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Experiment 02: Like experiment I, we have also checked out hidden measurements of 40, 80 , 160 and 40, 80 and 160 measurements of embedding. From Table III, the overall accuracy obtained for LSTM is 87% with a ratio of 90:10, embedding dimension 40, hidden dimensions 80 along with highest positive accuracy is 52% in 80:20 ratio and highest negative accuracy is 93% in 90:10 ratio.

Table 3: Accuracy Comparison with Word2VecEmbed Features for LSTM Technique.

Features	Train Test Ratio	Positive Accuracy	Negative Accuracy	Total Accuracy
Word2VecEmbed	90:10	50%	93%	87%
Word2VecEmbed	80:20	52%	89%	84%
Word2VecEmbed	70:30	51%	91%	85%

Experiment 03: In this experiment, we have tested the different ratio with 40, 80, 160 embedding dimensions and hidden dimensions of 40, 80 and 160. Here the best total accuracy is listed in ratio of 90:10 with embedding dimension 40 and hidden dimensions 80 which is listed in Table IV. Though the highest positive accuracy is in 80:20 ratio which is 53% and highest negative accuracy is 94% of 90:10 ratio which is also depicted in Table IV

Table 4: Accuracy Comparison with Fast text Embed Features for LSTM.

Features	Train Test Ratio	Positive Accuracy	Negative Accuracy	Total Accuracy
FastTextEmbed	90:10	50%	94%	84%
FastTextEmbed	80:20	53%	90%	82%
FastTextEmbed	70:30	51%	91%	83%

C. Result Analysis & Comparison

In this section, we examined our achieved results and compared them to some of the previous works. From Table V, we see the average accuracy of all three experiments result of LSTM technique and also allocated in the graphical representation of Figure 3. From these Table V and Figure 9

overall scenerio, we can compare that for Word2VecEmbed Features, LSTM perfroms better in total accuracy. Again, for positive accuracy, Trainable Embed Features perfom better than the others. On the other hand, FastText Embed Features outperform the others inspect of negative accuracy.

Table 5: Average Accuracy Comparison with Different Features for LSTM Technique

Features	Train Test Ratio	Positive Accuracy	Negative Accuracy	Total Accuracy
FastTextEmbed	90:10	58%	90%	83%
FastTextEmbed	80:20	51%	92%	85%
FastTextEmbed	70:30	51%	93%	84%

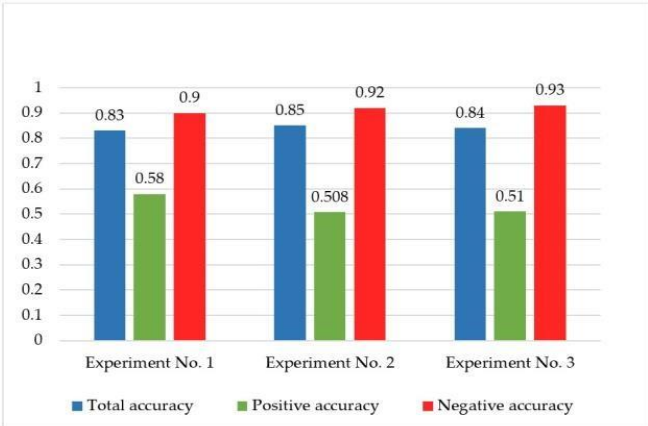

Figure 9: Variation of Average Accuracy over Different Features of LSTM Technique

Again, we compare our model with the existing of CNN technique where in Table VI, we can see the result of CNN of average accuracy of positive, negative and total. When we compare the result with our model, we see the difference and except for positive accuracy of ratio 80:20, LSTM is less accuracy than CNN. Other than that, in all ratio of accuracy, LSTM outperforms the state-of-art CNN technique. In Figure 9. There is a graphical representation of comparison between LSTM and CNN technique, where only average of total accuracy is depicted

Table 6: Average Accuracy Comparison With Different Features for CNN Technique

Features	Train Test Ratio	Positive Accuracy	Negative Accuracy	Total Accuracy
FastTextEmbed	90:10	48%	74%	72%
FastTextEmbed	80:20	52%	79%	68%
FastTextEmbed	70:30	49%	71%	69%

5. CONCLUSION

Sentiment mining from microblogging data has diverse applications in marketing, brand analytics, and research. Organizations use sentiment analysis to analyze the feedback from users about their products and services. Even the analysis of different domains can be used to conduct several types of research based on interest. Twitter, one of the most popular microblog, cover a variety of topics. In this paper, we have

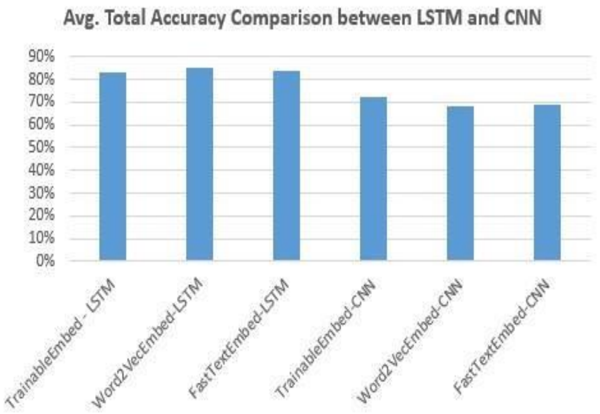

Figure 10: Comparison of Average Total Accuracy between LSTM and CNN Techniques

done the polarity of a public review whether it is positive or negative using a hybrid deep learning technique. We calculate the true positive rate and true negative rate from the mine of public reviews on twitter using various features. The challenge of this paper is that we calculate the positive and negative reviews, but the neutral sentence causes ambiguity. It is inappropriate for sarcasm sentences and emoticons like love, care, sad react is used on Facebook. In future work, we can work with different types of emoticons, and sarcasm reviews cause ambiguity for different media

6. FUTURE SCOPE

In future we have a plan to work with humor or sarcasm of Bangla text. We also work those Bangla sentence which are ambiguous. We have also work for emoticons.

REFERENCES

[1] E. Liddy, 'Natural Language Processing', *Sch. Inf. Stud. - Fac. Scholarsh.*, Jan. 2001, [Online]. Available: https://surface.syr.edu/istpub/63

[2] B. Pang and L. Lee, 'Opinion Mining and Sentiment Analysis', *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, Jan. 2008, doi: 10.1561/1500000011.

[3] B. Pang, L. Lee, and S. Vaithyanathan, 'Thumbs up? Sentiment Classification using Machine Learning Techniques', in *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, Jul. 2002, pp. 79–86. doi: 10.3115/1118693.1118704.

[4] T. M. V. Janssen, 'Frege, Contextuality and Compositionality', *J. Log. Lang. Inf.*, vol. 10, no. 1, pp. 115–136, 2001.

[5] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, 'Target-dependent Twitter Sentiment Classification', in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, USA, Jun. 2011, pp. 151–160. Accessed: Jul. 08, 2021. [Online]. Available: https://aclanthology.org/P11-1016

[6] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, 'Learning long-term dependencies in NARX recurrent neural networks', *IEEE Trans. Neural Netw.*, vol. 7, no. 6, pp. 1329–1338, Nov. 1996, doi: 10.1109/72.548162.

[7] X. Wang, Y. Liu, C. Sun, B. Wang, and X. Wang, 'Predicting Polarities of Tweets by Composing Word Embeddings with Long Short-Term Memory', in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China, Jul. 2015, pp. 1343–1353. doi: 10.3115/v1/P15-1130.

[8] D. Tang, B. Qin, and T. Liu, 'Document Modeling with Gated Recurrent Neural Network for Sentiment Classification', in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Sep. 2015, pp. 1422–1432. doi: 10.18653/v1/D15-1167.

[9] R. Socher *et al.*, 'Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank', in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA, Oct. 2013, pp. 1631–1642. Accessed: Jul. 08, 2021. [Online]. Available: https://aclanthology.org/D13-1170

[10] F. A. Gers, J. Schmidhuber, and F. Cummins, 'Learning to Forget: Continual Prediction with LSTM', *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000, doi: 10.1162/089976600300015015.

[11] 'Acoustic Modeling Using Deep Belief Networks'. https://ieeexplore.ieee.org/document/5704567 (accessed Jul. 08, 2021).

[12] S. Hochreiter and J. Schmidhuber, 'Long Short-Term Memory', *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[13] X. Zhou, X. Wan, and J. Xiao, 'Attention-based LSTM Network for Cross-Lingual Sentiment Classification', in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, Nov. 2016, pp. 247–256. doi: 10.18653/v1/D16-1024.

[14] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, 'A neural probabilistic language model', *J. Mach. Learn. Res.*, vol. 3, no. null, pp. 1137–1155, Mar. 2003.

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

[15] R. Collobert and J. Weston, ‘A unified architecture for natural language processing: deep neural networks with multitask learning’, in *Proceedings of the 25th international conference on Machine learning*, New York, NY, USA, Jul. 2008, pp. 160–167. doi: 10.1145/1390156.1390177.

[16] A. Yenter and A. Verma, ‘Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis’, in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, Oct. 2017, pp. 540–546. doi: 10.1109/UEMCON.2017.8249013.

[17] A. Dhar, H. Mukherjee, N. Sekhar Dash, and K. Roy, ‘Performance of Classifiers in Bangla Text Categorization’, in *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, Oct. 2018, pp. 168–173. doi: 10.1109/ICISET.2018.8745621.

[18] A. K. Mandal and R. Sen, ‘Supervised learning Methods for Bangla Web Document Categorization’, *ArXiv14102045 Cs*, Oct. 2014, Accessed: Jun. 12, 2021. [Online]. Available: http://arxiv.org/abs/1410.2045.

[19] Md. N. Hasan, S. Bhowmik, and Md. M. Rahaman, ‘Multi-label sentence classification using Bengali word embedding model’, in *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*, Dec. 2017, pp. 1–6. doi: 10.1109/EICT.2017.8275207.

[20] A. Rane and A. Kumar, ‘Sentiment Classification System of Twitter Data for US Airline Service Analysis’, in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, Jul. 2018, vol. 01, pp. 769–773. doi: 10.1109/COMPSAC.2018.00114.

[21] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, ‘A C-LSTM Neural Network for Text Classification’, *ArXiv151108630 Cs*, Nov. 2015, Accessed: Jul. 08, 2021. [Online]. Available: http://arxiv.org/abs/1511.08630

[22] F. B. F. L. F. T. M. H. is a stock analyst, W. with 10+ Y. of E. C. Stocks, markets, A. W. as Analyzing, and valuing companies L. about our editorial policies M. Hargrave, ‘How Deep Learning Can Help Prevent Financial Fraud’, *Investopedia*. https://www.investopedia.com/terms/d/deep-learning.asp (accessed Jul. 02, 2021).

[23] ‘What Is Deep Learning? | How It Works, Techniques & Applications’. https://www.mathworks.com/discovery/deep-learning.html (accessed Jul. 02, 2021).

[24] ‘What are Recurrent Neural Networks?’, Apr. 07, 2021. https://www.ibm.com/cloud/learn/recurrent-neural-networks (accessed Jul. 02, 2021).

[25] M. Venkatachalam, ‘Recurrent Neural Networks’, *Medium*, Jun. 22, 2019. https://towardsdatascience.com/recurrent-neural-networks-d4642c9bc7ce (accessed Jul. 02, 2021).

[26] ‘What is Natural Language Processing?’, Jun. 21, 2021. https://www.ibm.com/cloud/learn/natural-language-processing (accessed Jul. 09, 2021).

[27] T. Mikolov, K. Chen, G. S. Corrado, and J. A. Dean, ‘Computing numeric representations of words in a high-dimensional space’, US9740680B1, Aug. 22, 2017 Accessed: Jul. 09, 2021. [Online]. Available: https://patents.google.com/patent/US9740680B1/en

[28] ‘Gensim: topic modelling for humans’. https://radimrehurek.com/gensim/# (accessed Jul. 09, 2021).

[29] ‘Deep Learning | Introduction to Long Short Term Memory’, *GeeksforGeeks*, Jan. 16, 2019. https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/ (accessed Jun. 21, 2021).

[30] ‘Activation functions in Neural Networks’, *GeeksforGeeks*, Jan. 29, 2018. https://www.geeksforgeeks.org/activation-functions-neural-networks/ (accessed Jul. 09, 2021).

[31] ‘Activation Functions | Fundamentals Of Deep Learning’, *Analytics Vidhya*, Jan. 29, 2020. https://www.analyticsvidhya.com/blog/2020/01/fundamentals-deep-learning-activation-functions-when-to-use-them/ (accessed Jul. 09, 2021).

[32] ‘Sentiment Analysis - Twitter Dataset’. https://kaggle.com/seunowo/sentiment-analysis-twitter-dataset (accessed Jul. 09, 2021).

AUTHOR PROFILES

My name is **Md. Ahsan Habib**. I have completed my B.Sc (Engg.) from Mawlana Bhashani Science and Technology university Santosh-1902 Tangail, Dhaka, Bangladesh. I am also studying M.Sc(Engg.) on same university. Currently I am working as lecturer in University Grant Commission of Bangladesh(UGC) approved university which name is Bangladesh University which is located in Iqbal Road Mohammadpur, Dhaka, Bangladesh..