**An Intelligent Text to Emotion Twitter Sentiment Analysis Model using Bidirectional LSTM Networks**

1C.Jayapratha, 2V. Ramya, 3M. Ananda Kumar, 4R. Kalaiselvi,

1Professor and Head, Department of CSE, Karpaga Vinayaga College of Engineering and Technology, Chengelpet, Tamilnadu

2Assistant Professor, Department of ECE, Dhanalakshmi College of Engineering, Chennai Tamilnadu.

3,4Assistant Professor, Department of CSE, Arasu Engineering College, Kumbakonam, Tamilnadu.

1jayaprathaclement@gmail.com, 2ramya12vi@gmail.com, 3anandlogo@gmail.com, 4kalaiprathi85@gmail.com

**Abstract:** In the ever-evolving realm of social media platforms like Twitter, sentiment analysis has emerged as an indispensable instrument for fully comprehending the feelings of users. Through the utilization of Bidirectional LSTM Networks, this research proposes an intelligent Text to Emotion Twitter sentiment analysis model. This model integrates modern deep learning techniques with feature extraction methods in order to achieve improved performance. A thorough comprehension of user attitudes is made possible by the proposed model, which makes use of the Bi-LSTM architecture to capture both forward and backward contextual relationships in textual data. A Bag of Words (BoW) feature extraction approach is applied in order to maximize the input of the model. This technique creates structured numerical representations from unstructured textual data.

The Sentiment140 dataset, which is accessible to the general public and comprises 1.6 million tweets that have been categorized as either positive, negative, or neutral attitudes, was used to compile the dataset that was utilized in this investigation. It was necessary to do preprocessing on the dataset in order to guarantee that the input for feature extraction would be of a high quality. These preprocessing procedures included noise reduction, tokenization, and stemming. When compared to more conventional methods of machine learning, the results of the experiments show that the combination of the BoW methodology with the Bi-LSTM network leads in a considerable improvement in the accuracy of sentiment categorization. They also achieve superior results.

**Keywords: *Sentiment analysis, twitter data, social media, natural language, tokenization, stop words, long term, short memory and convolutional neural networks.***

1. **INTRODUCTION**

On social media platforms especially Twitter, people express their feelings, ideas, and sentiments on a variety of issues, which has resulted in the platforms being a rich source of textual data. For organizations, researchers, and governments, the extraction of relevant insights from this data is both a difficulty and an opportunity [1]. This is because of the enormous volume of material that is created every single day. Text is classified into emotional categories such as positive, negative, or neutral, which is a crucial component of sentiment analysis, which is a key feature of text mining. Analysis of social phenomena, monitoring of brand health, forecasting of trends, and understanding of public opinion are all aided by this method. Traditional machine learning techniques have given way to more complex deep learning methods over the course of time, each of which has improved the model's capacity to reliably identify human emotions that are encoded in text. It is through these changes that sentiment analysis has undergone a considerable evolution [2].

Naive Bayes, SVM and Decision Trees were some examples of an example of supervised learning algorithms that were used in traditional techniques of sentiment analysis. These models are able to predict outcomes based on predetermined characteristics and learn patterns from data that has been labeled. While the effectiveness of the model is strongly dependent on how the text is represented numerically, traditional techniques place a significant emphasis on feature extraction as a critical component [3]. For the purpose of transforming raw text into structured features, it was usual practice to employ methods such as Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), and n-grams. Even while these strategies are useful in a lot of situations, they have certain drawbacks.

Deep learning models have revolutionized sentiment analysis by automatically learning complex patterns and representations from raw data [4]. This is especially true for models that are built on neural networks of neural networks. Especially when it comes to dealing with the delicate intricacies of language, such as context, word order, and semantic links, these models perform exceptionally well in situations when standard techniques prove to be inadequate. Recurrent Neural Networks (RNNs), which are designed to analyze sequences of data such as phrases or full texts, are one of the most significant developments in deep learning for natural language processing (NLP) [5]. This is one of the most important advancements in the field. Because they are able to recall prior inputs and utilize this knowledge to impact future predictions, RNNs are particularly well-suited for the investigation of sentiment. When it comes to learning long-term dependencies, however, vanillas RNNs have limits. These restrictions frequently result in problems such as disappearing or expanding gradients on the network [6].

LSTM networks, which are a particular sort of RNN built to capture long-term dependencies in sequential data, were presented as a solution to these issues. LSTM networks are also known as Long Short-Term Memory networks. With the ability to remember information for extended periods of time, LSTM networks are exceptionally useful for comprehending the contextual connections that exist between the words in a phrase [7]. Because the meaning of a phrase may be determined by words that appear to be quite far apart, this skill is essential in the field of sentiment analysis. Although LSTMs provide considerable advances over typical RNNs, they still only process text in a single direction, either left-to-right or right-to-left. This restricts their capacity to completely comprehend the context in both directions at the same time that they are processing the text.

1. **LITERATURE SURVEY**

An area of research that examines people's subjective sentiments regarding products, services, organizations, personal events, subjects, qualities, and other things, such as opinions, emotional evaluations, perspectives, and attitudes, is called sentiment analysis. This field of study is sometimes referred to as opinion mining. Within the realm of Natural Language Processing (NLP), Text Sentiment Analysis (TSA) is among the most important undertakings [8]. Extraction of characteristics and classification of text based on emotion is a process. It has been discovered through a comparison of the existing models with superior performance that in order to successfully extract the context relations of sentences, it is necessary to divide sentences into word sequences and use the network model that is based on long-short memory or gated loop unit as the primary tool for processing sequence data. In recent times, scholars have increasingly been interested in sentiment analysis.

According to Phan et al. [9], a technique was developed that was based on combining the attitude towards a certain item from all tweets that were relevant to that object. It was proposed by Ahmed et al. [10] that a supervised neural weak model would be used with the intention of learning a collection of sentiment clusters embedding from the phrase global representation of the target domain. An approach to doing multimodal sentiment analysis that is based on deep learning was proposed by Sreevidya et al [11]. For the purpose of training sentiment word embedding, Peng et al. [12] suggested an adversarial learning technique. In this method, the discriminator was utilized to compel the generator to generate high-quality word embedding by making use of semantic and sentiment information.

For the purpose of improving the accuracy of sentiment classification, Tofighy and Fakhrahmad [13] suggested a statistical and context-aware feature reduction approach. To construct domain-specific sentiment lexicons (DL: Dynamic Lexicons), Mechulam et al. [14] developed a model that is based on a context-graph. This model may be utilized for the purpose of propagating the valence of a few seed words. In order to devise a strategy for suggesting smart phones, Kumar and Parimala [15] decided to make use of the sentiment analysis approach.

In their study, Kulkarni et al. [16] revealed that sentiment analysis is a potential method for quantifying consumer responses to branded viral video commercials. As a result, they proposed a typology of viral ad sharers that is based on sentiment research. During the process of writing the article, each of the publications provided us with a great deal of insight. On the basis of the references that were shown earlier in this article, we will provide a unique framework for the realization of the sentiment analysis activity.

1. **PROPOSED METHODOLOGIES**

The purpose of this part is to offer an alternative model for sentiment analysis, which we will refer to as Deep SA-Net, in order to analyze the sentiment of Twitter data in an easy and effective manner. For sentiment analysis, there are three different deep learning models that may be applied. These models are the 1D-CNN, the LSTM, and the Bi-LSTM model. The following procedure phases are included in the suggested methods:

* Dataset Collection
* Pre-processing the Data
* BoW Feature Extraction
* Building Bi-LSTM Model
* Training the Model
* Test Prediction

In Figure 1, the architecture of the SA-Net model that has been presented is displayed.

Deep Learning Model

Data Pre-processing

Tokenization

Normalization

Remove Pinc, digit, char, stop words

Stemming

Twitter Train Dataset

BoW Feature Extraction

Bi-LSTM

Train

Data

Pre-processing

Twitter Test Dataset

**Fig. 1** Architecture of Text to Emotion Net Model

* 1. **Dataset Collection**

The Twitter\_training.csv dataset, which is accessible to the general public, was developed for the purpose of NLP and SA, with a primary emphasis on Twitter data [17]. In most cases, it consists of a number of columns, such as Tweet ID, which is a one-of-a-kind identifier for each tweet, and Text, which is the actual text of the tweet itself. A Sentiment column is also included in the dataset. As shown in Figure 2, this column represents the label or sentiment that is connected with the tweet, such as Positive, Negative, and Neutral.



**Fig. 2** The sample dataset descriptions

A few examples of other components that are not required are the timestamp, which offers information about the time at which the tweet was sent, the user ID, which identifies the person who sent the tweet, and the location, which provides information about the geographical attributes of the place. In certain instances, the dataset could also contain a Language column that specifies the language that the tweet was written in. The dataset information of twitter data, number of classes and number of records in each class is shown in Figure 3.

**Fig. 3** Dataset information of twitter record

* 1. **Pre-processing the Dataset**

The first step in the architecture is to pre-process raw Twitter data, as it often contains irrelevant and noisy information. Pre-processing ensures the text is clean, structured, and ready for feature extraction. The pre-processing consist of the following steps:

**Tokenization:** Breaking the tweets into smaller units (tokens) such as words or phrases for further processing.

**Normalization:** Standardizing text by converting all characters to lowercase, handling repeated characters, and ensuring consistency.

**Removal of Noise:** Eliminating unnecessary elements such as punctuation, digits, special characters, and stop words (common words that do not contribute to the sentiment).

Stemming: Reducing words to their root forms to unify similar words (e.g., "playing" and "played" are reduced to "play").

**Stop Words:** Removing frequently occurring words such as "and," "the," or "is," which do not carry significant sentiment.

**Stemming:** Words are reduced to their root forms to unify similar words into a single representation. For example, "running," "runs," and "ran" are stemmed to “run”.

* 1. **BoW Feature Extraction**

After pre-processing, the text data is converted into numerical features, which serve as inputs to the deep learning model. The Bag of Words (BoW) technique is employed for extracting meaningful features from the twitter data. In this technique, the text is represented as a vector of word frequencies or binary indicators. For instance, the sentence "AI is great" might be transformed into a vector like [1, 1, 0, 1], indicating the presence or absence of specific words in the vocabulary. BoW captures the importance of individual words in the dataset but does not retain information about the order of words. While simple, this method is effective in representing text for sentiment analysis tasks.

* 1. **Bi-LSTM**

Bi-LSTM is a specialized type of neural network designed to process sequential data, such as text, speech, and time-series data, in both forward and backward directions. Unlike a traditional LSTM, which only processes data in one direction, Bi-LSTM incorporates two LSTM layers: one processes the sequence from the beginning to the end, and the other processes it from the end to the beginning. This bidirectional approach enables the model to utilize both past and future contexts, improving its ability to understand sequences.



**Fig. 4** Architecture of Bi-LSTM Model

The core idea behind LSTM, and by extension Bi-LSTM, is its memory cell, which retains information over long time periods. This memory cell is regulated by three gates: the forget gate decides what information to discard, the input gate determines what new information to store, and the output gate decides what information to pass to the next layer or time step. These gates help address the vanishing gradient problem, a common issue in traditional RNNs, allowing LSTM to learn long-term dependencies. Figure 4 shows the first Bi-LSTM model built using basic input features.

* 1. **Training the Model**

The training phase of a Bi-LSTM model involves teaching the model to understand patterns in the input data. First, the data is cleaned and converted into a format the model can understand, like numbers or word embeddings. Then, the Bi-LSTM processes the data in two directions—one from start to end and the other from end to start. The model compares its predictions with the correct answers and calculates the error using a loss function. This error is used to adjust the model’s weights through a process called backpropagation, making it better at predicting over time. Optimization algorithms like Adam or RMSprop help speed up the learning process. To prevent the model from overfitting or memorizing the data, techniques dropout are used. The training happens in multiple epochs, and the model's performance is tested on a validation dataset to ensure it is improving and not overfitting. The summary of trainable, non-trainable parameter of Bi-LSTM models is shown in Figure 5.



**Fig. 5** Trainable, non-trainable parameter of Bi-LSTM model

1. **EXPERIMENTAL RESULTS AND DISCUSSIONS**

In this section, we focus on the experimental setup, how the Bi-LSTM model performed, and the results obtained after training and testing on specific tasks like sentiment analysis, text classification, or any sequential data processing task. Using standard deep learning methods, we have assessed the suggested efficient Text to Emotion sentiment analysis model for the twitter data. A system with an i5 CPU, 8 GB of RAM, and 1 TB of hard drive was used to construct the suggested model using the Anaconda IDE tools and Python.

* 1. **Performance Metrics**

**Prec.:** Precision is used to calculate the proportion of correctly predicted instances over all predictions. The precision can be measured as follow:

$Prec.=\frac{tp}{tp+fp}$ (2)

**Rec.:** Recall is the proportion of correctly predicted occurrences over all instances.

$Rec.=\frac{tp}{tp+fn}$ (3)

**Acc.:** The accuracy measure is calculated by taking the total number of input samples and dividing it by the number of correct predictions to get.

$Acc.=\frac{tp+fp}{tp+fp+tn+fn}$ (4)

**F1-Sc.:** Using the F1-measure, helps to strike a compromise between the accuracy and recall measures. For the purpose of calculating the F1- score, the following formula can be utilized:

$F=2×\frac{Prec.×Rec.}{Prec.+Rec.}$ (5)

* 1. **Result and Discussions**

In this section, we conducted a detailed evaluation of text to emotion Bi-LSTM sentiment analysis model to assess their performance on the given twitter sentiment analysis dataset. For model, critical performance metrics were computed and summarized in Table 1 and Figure 5. The Bi-LSTM model emerged as the best-performing model, achieving the highest classification accuracy of 95%, along with commendable scores across the other metrics, highlighting its robustness in distinguishing between classes.

**Table 1** Performance analysis of SA-Net sentiment analysis model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S. No.** | **Metrics** | **Negative** | **Neutral** | **Positive** |
| **1.** | Precision | 94.85 | 98.23 | 97.07 |
| **2.** | Recall | 93.32 | 97.04 | 92.99 |
| **3.** | F1-Score | 94.08 | 97.63 | 95.02 |
| **4.** | Accuracy | 95.14 | 95.14 | 95.14 |

**Fig. 5** Performance analysis of text to emotion model

In addition, the confusion matrix of the text to emotion Bi-LSTM sentiment analysis model, which can be seen in Figures 6, is presented below.



**Fig. 6** CM oftext to emotion Bi-LSTM Model

The model was trained for a 20 epochs and the learning rate was adjusted using an adaptive optimizer (Adam). Regularization techniques like dropout were used to prevent overfitting. The model was validated on a separate validation set to tune hyperparameters and monitor performance during training. The performance training accuracy, validation accuracy and losses of text to emotion Bi-LSTM sentiment analysis model is shown in Figure 7.

**Fig. 7** Bi-LSTM text to emotion model accuracy and loss

After training, the model was tested on a hold-out test dataset, which was not seen during the training phase. The predictions made by the Bi-LSTM model were compared with the actual labels in the test data to compute evaluation metrics. The sample test prediction of text to emotion Bi-LSTM model is shown in Figure 8.

****

**Fig. 8** Test prediction ofBi-LSTM text to emotion model

1. **CONCLUSION**

This study presents an effective solution for sentiment analysis on Twitter data by integrating Bidirectional LSTM Networks with Bag of Words (BoW) feature extraction. By leveraging Bi-LSTM's ability to capture both forward and backward contextual dependencies in textual data, the proposed model provides a comprehensive understanding of user sentiments. The use of BoW for feature extraction further enhances the model's ability to transform unstructured textual data into meaningful numerical representations, leading to improved performance. Experimental results using the Sentiment140 dataset demonstrate that the combined approach outperforms traditional machine learning methods, achieving higher accuracy 95.5% in sentiment classification.

**REFERENCES**

1. A. Shoukry and A. Rafea, “Sentence-level Arabic sentiment analysis,” in Proceedings of the 2012 International Conference on Collaboration Technologies and Systems, CTS 2012, 2012, pp. 546–550, doi: 10.1109/CTS.2012.6261103.
2. R. Duwairi and M. El-Orfali, “A study of the effects of preprocessing strategies on sentiment analysis for Arabic text,” J. Inf. Sci., vol. 40, no. 4, pp. 501–513, 2014, doi: 10.1177/0165551514534143.
3. Alonso, M.A. et al.(2021) ‘Sentiment Analysis for Fake News Detection’, Electronics, 10(11), p. 1348. Available at: <https://doi.org/10.3390/electronics10111348>.
4. Chaubey, P.K. et al.(2022) ‘Sentiment Analysis of Image with Text Caption using Deep Learning Techniques’, Computational Intelligence and Neuroscience. Edited by V. Kumar, 2022, pp. 1–11.
5. P.Deepan and L.R. Sudha, “Deep Learning and its Applications related to IoT and Computer Vision”, Artificial Intelligence and IoT: Smart Convergence for Eco-friendly Topography, Springer Nature, pp. 223-244, 2021, <https://doi.org/10.1007/978-981-33-6400-4_11>.
6. Lin, Y.; Li, J.P.; Yang, L.; Xu, K.; Lin, H.F. Sentiment Analysis With Comparison Enhanced Deep Neural Network. IEEE Access. 2020, 8, 78378-78384.
7. Ghantasala, G. P., Sudha, L. R., Priya, T. V., Deepan, P., & Vignesh, R. R. An Efficient Deep Learning Framework for Multimedia Big Data Analytics. Multimedia Computing Systems and Virtual Reality, 99.
8. Asif, M.; Ishtiaq, A.; Ahmad, H.; Aljuaid, H.; Shah,J. Sentiment analysis of extremism in social media from textual information. Telematics & Informatics 2020, 48: 101345.
9. Phan, H.T.; Nguyen, N.T.; Tran, V.C.;Hwang, D.; Szczerbicki, E.;Trawi ń ski, B. A sentiment analysis method of objects by integrating sentiments from tweets. Journal of Intelligent & Fuzzy Systems. 2019, 37(6), 7251-7263.
10. Ahmed, M.; Chen, Q.; Li, Z. Constructing domain-dependent sentiment dictionary for sentiment analysis. Neural Computing and Applications 2020, 10.1007/s00521-020-04824-8.
11. Sreevidya, P.; Murthy, O. V. R.; Veni, S. Sentiment analysis by deep learning approaches. Telkomnika, 2020, 18(2), 752-760.
12. Peng, B.; Wang,J.; Zhang, X. Adversarial learning of sentiment word representations for sentiment analysis. Information Sciences 2020, 541: 426-441.
13. Tofighy, S.; Fakhrahmad, S.M. A proposed scheme for sentiment analysis. Kybernetes, 2018, 47(5), 957-984.
14. Mechulam, N.; Salvia, D.; Ros, A.; Etcheverry, M. Building Dynamic lexicons for sentiment analysis. Inteligencia Artificial: Revista Iberoamericana de Inteligencia Artificial, 2019, 22(64), 1-13.
15. Kumar, G.; Parimala, N. An integration of sentiment analysis and MCDM approach for smartphone recommendation. International Journal of Information Technology & Decision Making. 2020, 19(4), 1037-1063.
16. Kulkarni, K.K.; Kalro, A.D.; Sharma, D.; Sharma, P. A typology of viral ad sharers using sentiment analysis. Journal of Retailing and Consumer Services. 2020, 53, doi:10.1016/j.jretconser.2019.01.008.
17. <https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis>