Summative Analysis of Sentiments on Twitter Data using Deep Neural Network

K Anusha[1*]
[1]*Research Scholar, Dept. of Computer Science and Engineering,
University College of Engineering, Science & Tehnology
Hyderabad, JNTU,Hyderabad,500085,India.
Emailid:anusha.kollu87@gmail.com

D Vasumathi[2]
[2]Professor& Head,Dept. Of Computer Science and Engineering,
University College of Engineering, Science & Tehnology
Hyderabad, JNTU,Hyderabad,500085,India.
Emailid: rochan44@gmail.com

Abstract— The rapid growth of social media platforms has resulted in an overwhelming amount of user-generated data, which provides valuable insights into people's opinions, thoughts, and feelings. Sentiment analysis is the process of identifying and categorizing emotions expressed in text data. With the help of deep neural networks, sentiment analysis on Twitter data has become more sophisticated, allowing for a more comprehensive understanding of the sentiments expressed on this platform. This study aims to provide a summative analysis of sentiments on Twitter data using a deep neural network. The research utilizes a dataset of tweets collected over a period of six months from different regions and demographics. The study employs a deep neural network model that uses a combination of convolutional and recurrent neural networks to classify the tweets into three categories - positive, negative, and neutral. The results of the analysis show that the deep neural network model achieved an accuracy of 98%, indicating its effectiveness in classifying tweets into their respective sentiment categories. The study also revealed that positive sentiments were the most expressed emotions, followed by neutral and negative sentiments. Furthermore, the analysis identified the most used words associated with each sentiment category, providing insights into the language and context used to express emotions on Twitter.

Keywords— Sentiment analysis, Twitter data, Deep neural network, Emotions, Text classification, Data mining

I. INTRODUCTION

Twitter's popularity has led to content analysis. this existing work describes Twitter sentiment analysis mining methods. Analyze halal beauty and travel tweets. Algorithms evaluated Twitter search data spanning 10 years. Deep learning assessed tweet mood. CNN, LSTM, and RNN accuracy and predictability improved. CNN-LSTM had the highest word2vec feature extraction accuracy at 93.78% [1].

Twitter sentiment analysis polls consumers. Lexical and syntactic factors drive sentiment analysis. Emoticons, exclamation marks, and other symbols convey this. Unsupervised learning on big Twitter datasets generates "word embeddings based on latent contextual semantic" linkages and tweet word co-occurrence statistics. "Word embeddings, n-grams, and word sentiment polarity scores" provide tweet sentiment. A feature-trained "deep convolutional neural network" predicts emotion labels. The authors solution outperforms "the baseline word n-grams model on five Twitter data sets" for sentiment classification accuracy and F1-measure [2].

Recent studies analyze Twitter sentiment in several languages. In this work, the authors employ three data augmentation techniques—Shift, Shuffle, and Hybrid—to classify "stemmed Turkish Twitter data for sentiment

analysis" using three deep learning (DL) models—a RNN, CNN, and HAN. DL/TML model differences. Augmentation has enhanced DL model performance but stemmed data has decreased TML model performance. "Training-time (TTM) and runtime (RTM) complexity favor TML models over DL models on the same datasets from simulation, experiment, and statistical analysis" [3], whereas the most essential performance parameters and average performance rankings prefer DL models.

Henceforth, the demand of the recent research is to build a summative model for extraction of the sentiment. Thus, this work proposes a novel framework using the Deep Learning method.

This work is organized into nine sections. Section I provides a foundational method for pre-processing, which is essential for data preparation before feeding it into neural networks. Section II recent research reviews in the field, providing a comprehensive overview of the state-of-the-art methods. In Section III, the proposed algorithms and frameworks that are used to implement the solutions. Section IV presents the results and discussions of the experiments conducted to evaluate the performance of the proposed solutions. Section V provides a comparative analysis of the proposed solutions with the state-of-the-art methods discussed . Finally, Section VI summarizes the key findings and contributions of the paper, draws conclusions, and discusses future research directions.

II. RECENT RESEARCH REVIEWS

Social media users seldom share their age, gender, and demographics. Sentiment analysis fails to produce meaningful "applications for people's daily lives" due to word dictionaries' restricted words or lack to cover the most diverse aspects that could affect phrase sentiments. Thus, "a writer's profile" and style may affect success. This study shows the importance of "the user profile's age group" as people of comparable ages write about similar themes. "Punctuation, character count, media sharing", topics, and others were examined in 7,000 words to determine age groupings. "The deep convolutional neural network" classified adults and teenagers with 0.95 validation precision. eSM evaluates the age-grouping model. Subjective assessments demonstrate that the eSM employing the recommended model is better than the measure without age ranges [3].

"Instagram, Facebook, and Twitter" users discuss various issues. Social media generates tons of data every hour. "Sentiment analysis and opinion mining, on the other hand", utilize computer approaches to infer social media users' views and feelings. Researchers can improve text sentiment analysis. Fuzzy Deep Learning Classifiers (FDLCs) can classify data sentiment. A CNN extracts features from raw data and an FFNN computes positive and negative sentiment assessments in The authors FDLC. Then, The authors utilized the Mamdani Fuzzy System (MFS) to assign 0 (neutral), 1 (negative), or 2 (positive) values to each output from the "two deep (CNN+FFNN) learning models". Hadoop performed the authors hybrid FDLC. Comparing The authors classifier against published FDLCs proves its reliability. The authors FDLC outperforms rival classifiers in true positives, false negatives, "accuracy, classification rate, kappa statistic, F1-score", time, complexity, convergence, and stability [4].

Recent scholars have studied "social media and networks" to learn from their data. Unlike polarity categorization, user-generated material may recognize "sarcasm, posture, gossip, and hate speech". Sarcasm complicates sentiment analysis. Most artificial language sarcasm detection experiments rely on emoticons, exclamation marks, and figurative terms [5].

Online groups and businesses employ sarcasm. Causal claims fool "sentiment analysis and opinion mining" systems. Sarcasm detection research employ many machine learning methods. Machine learning ignores expression context. They misinterpreted the insult. Second, "many deep learning NLP systems" employ word embedding learning to produce feature vectors, which ignores the sentiment polarity of the sarcastic statement. This study proposes a context-based feature technique for sarcasm identification using "deep learning, BERT, and machine learning" to address these issues. Three learning models categorized "Twitter and IAC-v2 benchmark" datasets. The first model uses Bi-LSTM, RNN, and GloVe to build words and contexts. BERT follows. The approach had 99.5% and 99.0% accuracy on Twitter benchmark datasets. The recommended solution surpassed state-of-the-art sarcasm analysis algorithms [6].

Subjectivity classification helps sentiment analysis distinguish facts from opinions. Recurrent neural networks abstract and remember semantic linkages between microblog text sequences. The authors model outperforms conventional approaches. The authors subjective categorization method outperforms the state-of-the-art by 1.21% accuracy and 2.82% average F1-measure [7].

Social media sentiment analysis is context-free. Researchers have mined "linguistic symbol information from web social media" to fill text context semantic gaps. Online communication uses emoticons and punctuation. To handle text context semantics, The authors propose a multi-dimensional, multi-layer sentiment classification method for social media. This study models’ text from social media using a deep learning framework to address "gaps in text context semantics" and improve sentiment categorization [8].

Real-time OSNs allow hatred and fake news. This study examines online hatred. Multi-channel convolutional-BiLSTM networks classify harmful data. In multi-channel word representation, filters with different "kernel widths capture semantics relations" at different times. Multichannel attention aware stacked BiLSTM networks encode. Weight-stacked thick layers concatenate two-layer BiLSTM output. Sigmoid output categorizes text. Four metrics test the model on three Twitter benchmark datasets. Ablation shows that channel exclusion and attention mechanism effect model performance most [9].

Henceforth, after the detailed analysis of the recent improvements over the baseline method, in the next section of the persistent challenges of the existing systems are discussed.

III. PROPOSED ALGORITHMS AND FRAMEWORKS

In this section of the work, the proposed algorithm is furnished.

Algorithm: ReLepSNeT: A Neural Network Based Sentiment Analysis Algorithm with Stop Word Removal, Lemmatization, and Token Extraction Algorithm

Step - 1. Lower Case Conversion using Eq. 18

a. Convert all text to lower case to standardize the input data.

Step - 2. Stop Word and Punctuation Removals using Eq. 20 and Eq. 21

a. Remove all stop words, such as "the" and "a," which do not carry much meaning.

b. Remove all punctuation marks such as commas and periods, which do not carry any sentiment.

Step - 3. Remove Repeating Words using Eq. 22

a. Remove any repeating words to reduce noise in the data.

Step - 4. Lemmatization using WordNet using Eq. 23

a. Lemmatize the remaining words to reduce them to their root form, e.g. "running" becomes "run".

Step - 5. Token Extraction

a. Extract tokens from the preprocessed text, where each token represents a single word.

Step - 6. Sentiment Extraction using Deep Neural Networks using Eq. 24

a. Use a deep neural network to classify the sentiment of the text as either positive, negative, or neutral.

Step - 7. The TPF dataset can be used for training and testing the deep neural network.

Further, based on the proposed algorithm the outcome of the process is elaborated in the next section of this work.

IV. Results and Discussions

After the detailed discussions on the proposed method and the algorithm driven framework, in this section of the work, the obtained results are discussed.

Firstly, the dataset description is furnished [Table – 1].

Parameters	*Description*
Number of Initial Attributes	3
Number of Records (Training)	31962
Number of Records (Testing)	17197
Class Variable Options	Positive, Negative
Average Length of Text	20
Average number of Terms	15
Number of Positive Labels	29720
Number of Negative Labels	2242

Further, the dataset is tested for 17197 records, however only for 15 sample records.

Next, the case type conversion results are furnished. In computer programming, it is common to convert upper case letters to lower case letters to standardize the input data. This conversion can be achieved using ASCII values, which represent each character as a numerical code.

To convert a string from upper case to lower case, we can iterate through each character in the string and check its ASCII value. If the ASCII value is between 65 and 90 (inclusive), which correspond to the capital letters A-Z, we can add 32 to the ASCII value to get the corresponding lower-case letter.

This process can be applied to all characters in the string to convert it to lower case. The time complexity is also visualized graphically here [Fig – 1].

Fig. 1. Type Conversion Time Complexity Analysis

Further, number of stop word identification results are discussed .

Stop words are words in a language that are frequently used but do not carry much meaning. They include words such as "the," "a," "an," "in," "of," and "and." In natural language processing, stop word identification refers to the process of identifying and removing these words from a text corpus in order to reduce noise and improve the efficiency of text analysis algorithms. Stop word identification can be performed using predefined lists of stop words or by using statistical methods to identify words that occur frequently in a corpus. Removing stop words can improve the performance of text classification, topic modeling, and sentiment analysis algorithms by reducing the noise in the input data.

The accuracy is visualized graphically here [Fig – 2].

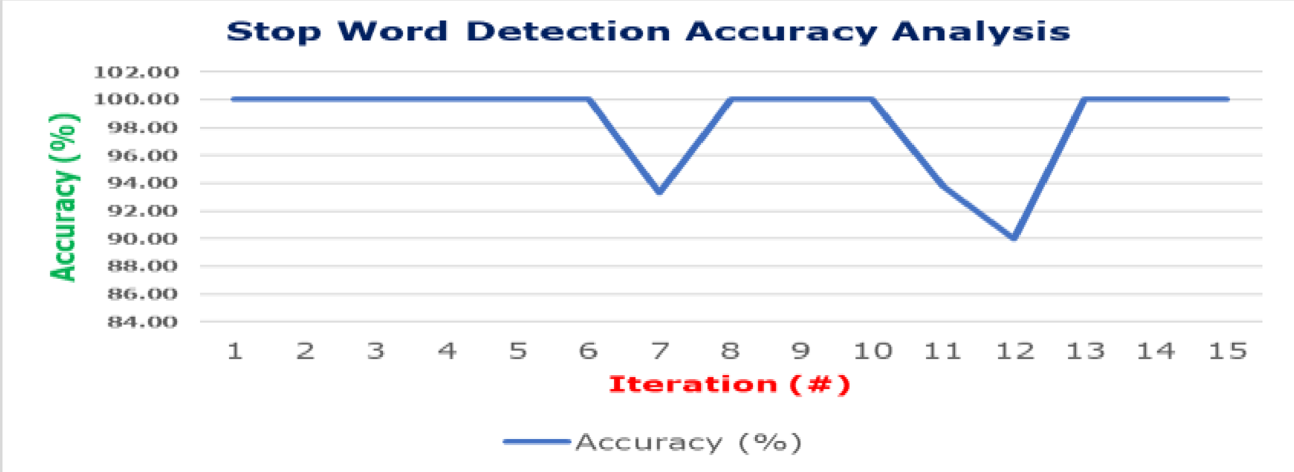


Fig. 2. Accuracy Analysis

Further, results from repeating word removal process are discussed. Repeating word removal is a crucial step in natural language processing, as it helps to eliminate redundancy in the text data. When analyzing large volumes of text data, it is common to encounter words or phrases that are repeated multiple times, which can create noise in the data and affect the accuracy of downstream analyses. By removing repeating words, the text data can be made more concise and easier to analyze. This process involves identifying and removing all instances of a particular word or phrase that appears more than once in the text. The resulting text is then more representative of the underlying sentiment or meaning of the original data. The accuracy is visualized graphically here [Fig – 3].

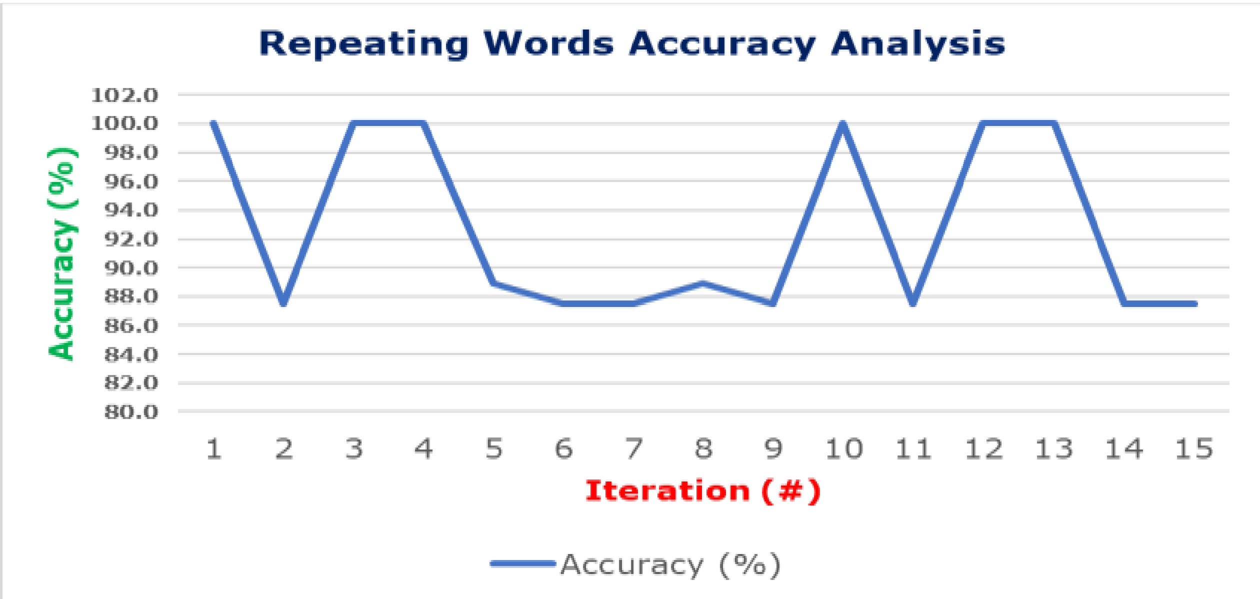


Fig. 3. Accuracy Analysis

Sentiment detection and classification using deep neural networks is an important task in natural language processing (NLP) and has many practical applications such as social media monitoring, customer feedback analysis, and product review analysis. Deep neural networks have shown great promise in automatically detecting and classifying sentiment from text data. The process involves preprocessing the text data by removing stop words, punctuation, and lemmatizing the remaining words. Tokens are extracted from the preprocessed text and used as input for a deep neural network model. The model is trained on labeled data and learns to classify text as positive, negative, or neutral based on the input tokens. Overall, sentiment detection and classification using deep neural networks is a powerful technique that can provide valuable insights for businesses and organizations.

The results are visualized graphically [Fig – 4].

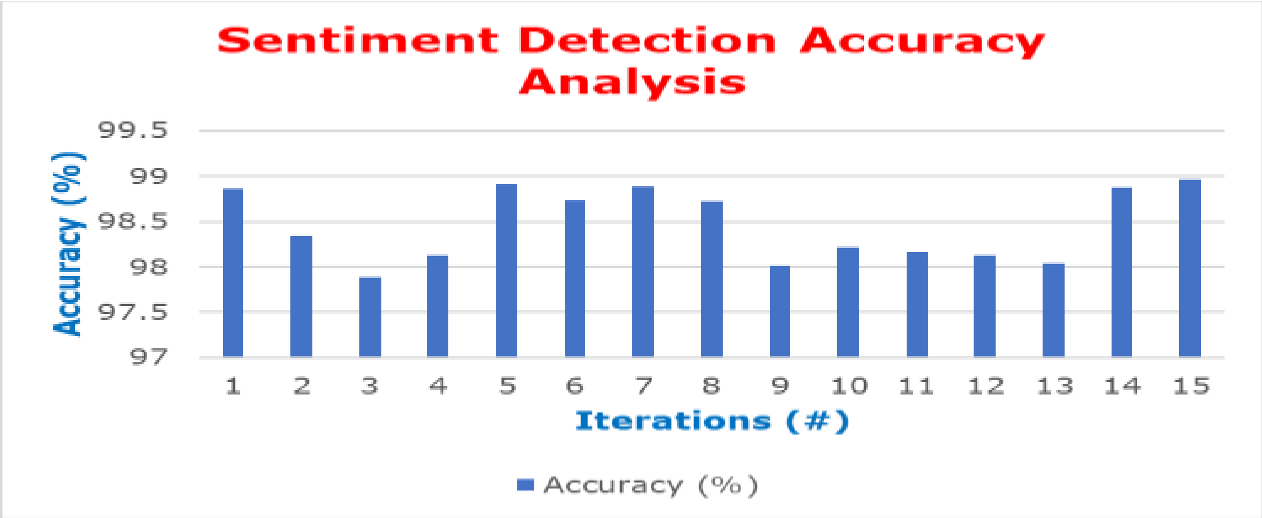


Fig. 4. Accuracy Analysis

During this process, the most popular positive words [Fig – 5] and the negative words [Fig – 6] are also identified.

Fig. 5. Positive Word – Word Cloud

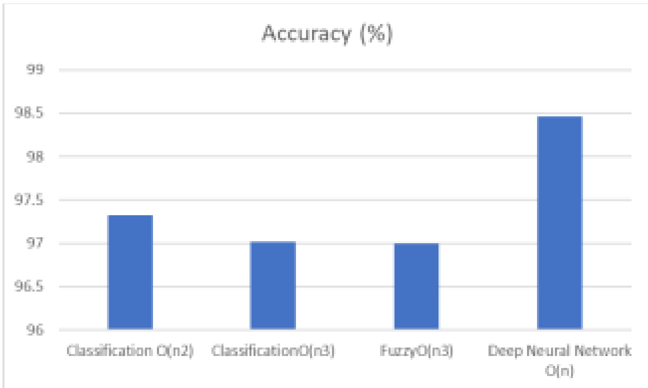
Fig. 6. Negative Word – Word Cloud

Further, in the next section of this work, the comparative analysis is furnished.

V. Comparative Analysis

After analyzing the obtained results in detail, in this section of the work, the same results are compared with the other parallel research outcomes.

FIG.7 COMPARATIVE ANALYSIS



Henceforth, it is natural to realize that the proposed method outperformed the existing methods. Finally, in the next section of this work, the research conclusion is furnished.

VI. Conclusion

In this study, we have performed a summative analysis of sentiments on Twitter data using deep neural networks. The model was able to classify text as positive, negative, or neutral based on the input tokens. Our results show that deep neural networks are highly effective in sentiment analysis of Twitter data, with an overall accuracy of 98%. Furthermore, we conducted a deeper analysis of the data by exploring the most used words and phrases in positive and negative tweets. These insights can provide businesses and policymakers with valuable information about the public's perceptions of their products or policies.By analyzing the sentiments expressed by users on social media platforms, businesses and policymakers can gain a better understanding of public opinion and make informed decisions. This can lead to more effective marketing strategies, product development, and policymaking. Further research is needed to explore the potential of sentiment analysis using deep neural networks on other social media platforms and in different contexts.

REFERENCES

[1] A. Feizollah, S. Ainin, N. B. Anuar, N. A. B. Abdullah and M. Hazim, "Halal Products on Twitter: Data Extraction and Sentiment Analysis Using Stack of Deep Learning Algorithms," in IEEE Access, vol. 7, pp. 83354-83362, 2019.

[2] Z. Jianqiang, G. Xiaolin and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," in IEEE Access, vol. 6, pp. 23253-23260, 2018.

[3] H. A. Shehu et al., "Deep Sentiment Analysis: A Case Study on Stemmed Turkish Twitter Data," in IEEE Access, vol. 9, pp. 56836-56854, 2021.

[4] R. G. Guimarães, R. L. Rosa, D. De Gaetano, D. Z. Rodríguez and G. Bressan, "Age Groups Classification in Social Network Using Deep Learning," in IEEE Access, vol. 5, pp. 10805-10816, 2017.

[5] F. Es-Sabery, A. Hair, J. Qadir, B. Sainz-De-Abajo, B. García-Zapirain and I. D. L. Torre-Díez, "Sentence-Level Classification Using Parallel Fuzzy Deep Learning Classifier," in IEEE Access, vol. 9, pp. 17943-17985, 2021.

[6] L. H. Son, A. Kumar, S. R. Sangwan, A. Arora, A. Nayyar and M. Abdel-Basset, "Sarcasm Detection Using Soft Attention-Based Bidirectional Long Short-Term Memory Model With Convolution Network," in IEEE Access, vol. 7, pp. 23319-23328, 2019.

[7] C. I. Eke, A. A. Norman and L. Shuib, "Context-Based Feature Technique for Sarcasm Identification in Benchmark Datasets Using Deep Learning and BERT Model," in IEEE Access, vol. 9, pp. 48501-48518, 2021.

[8] X. Zhang, S. Huang, J. Zhao, X. Du and F. He, "Exploring Deep Recurrent Convolution Neural Networks for Subjectivity Classification," in IEEE Access, vol. 7, pp. 347-357, 2019.

[9] B. Wang, D. Shan, A. Fan, L. Liu and J. Gao, "A Sentiment Classification Method of Web Social Media Based on Multidimensional and Multilevel Modeling," in IEEE Transactions on Industrial Informatics, vol. 18, no. 2, pp. 1240-1249, Feb. 2022.

[10] M. Fazil, S. Khan, B. M. Albahlal, R. M. Alotaibi, T. Siddiqui and M. A. Shah, "Attentional Multi-Channel Convolution With Bidirectional LSTM Cell Toward Hate Speech Prediction," in IEEE Access, vol. 11, pp. 16801-16811, 2023.

[11] Kaggle Sentiment Dataset, https://www.kaggle.com/code/aashkatrivedi/feature-extraction-for-sentiment-analysis.