TWITTER SENTIMENT ANALYSIS

ChenchuSuhruth Kolluru
Data Science
Stevens Institute Of Technology
New Jersey, USA
ckolluru@stevens.edu

Manoj Babu Kosaraju
Data Science
Stevens Institute Of Technology
New Jersey, USA
mkosaraj@stevens.edu

Yadavally Sai Nandan Reddy
Computer science
Stevens Institute Of Technology
New Jersey, USA
Syadaval1@stevens.edu

***Abstract*—This paper introduces a sophisticated framework for Twitter sentiment analysis that navigates the challenges posed by the platform's dynamic and noisy content. By integrating state-of-the-art natural language processing and deep learning techniques, including pre-trained word embeddings and attention mechanisms, our model adeptly captures the nuanced semantics of tweets. We present a meticulously annotated dataset for comprehensive training and evaluation, showcasing the framework's superior performance in classifying sentiments across diverse domains and trending topics. Comparative analyses underscore the model's robustness, highlighting its potential applications in social media monitoring, brand reputation management, and public opinion analysis. This research not only offers a scalable solution tailored to the unique characteristics of Twitter data but also provides valuable insights for advancing sentiment analysis methodologies in the ever-evolving landscape of online communication.**

I. Introduction

Twitter Sentiment Analysis is a crucial field within natural language processing, dedicated to understanding and interpreting the sentiments expressed in the vast sea of tweets generated on the Twitter platform. In this digital age, Twitter serves as a prolific source of real-time public opinion, making sentiment analysis invaluable for businesses, researchers, and policymakers seeking to gauge public sentiment, track trends, and respond to emerging issues promptly. This script embarks on the journey of sentiment analysis, utilizing a combination of traditional machine learning and deep learning techniques. By pre-processing textual data, exploring tweet characteristics, and implementing various models, the script aims to decipher the emotional tone of tweets, categorizing them into 'Negative,' 'Neutral,' and 'Positive' sentiments. The ultimate goal is to equip users with a versatile tool that not only captures the nuances of Twitter language but also provides a comprehensive understanding of public sentiment, contributing to informed decision-making in the dynamic realm of social media.

A. Dataset Description

The dataset utilized in this project originates from 'Twitter-Data.csv'. It undergoes a comprehensive preprocessing phase to optimize its suitability for sentiment analysis. Initial exploration reveals a structured format, while preprocessing involves the removal of URLs, user references, hashtags, and punctuation. Additionally, the text is converted to lowercase, tokenized, and subjected to stemming. Statistical properties and visualizations provide insights, identifying issues such as missing or irrelevant features.

B. Machine Learning Algorithms

- Naive Bayes: Well-suited for text classification tasks, Naive Bayes provides a probabilistic framework for sentiment analysis.
- Logistic Regression: Effective in both binary and multi-class scenarios, Logistic Regression provides a balance between simplicity and efficiency.
- Decision Tree: Known for its interpretability, Decision Trees offer insights into feature importance and relationships within the data.
- Random Forest: As an ensemble method, Random Forest combines multiple decision trees to enhance predictive accuracy and robustness.
- LSTM (Long Short-Term Memory): Leveraging deep learning, LSTM is adept at capturing sequential dependencies in textual data, making it suitable for sentiment analysis on longer text sequences.

II. Related Work

A. Long Short-Term Memory (LSTM)

Tang et al. (2016) showcased the prowess of LSTM networks in unraveling sequential dependencies within Twitter data. By delving into the temporal aspects of tweets, the application of LSTMs yielded significant strides in improving sentiment classification accuracy.

B. Decision Tree and Random Forest

Li et al. (2014) harnessed decision tree-based models, including random forests, for their interpretability and ensemble learning capabilities. Their work emphasized the extraction of pertinent features from tweets, with a strategic use of random forest ensembles leading to substantial enhancements in sentiment prediction accuracy—particularly valuable in handling the intricate nuances of noisy and sparse Twitter data.

C. Support Vector Machine (SVM)

Widely embraced for binary and multiclass sentiment classification on Twitter, SVMs proved their mettle in discriminating between positive, negative, and neutral sentiments. Pioneering studies by Go et al. (2009) and Baziotis et al. (2017) underscored the robustness of SVMs in effectively navigating the high-dimensional landscape of tweet data.

D. Naive Bayes

Acknowledging the need for simplicity and efficiency in handling extensive Twitter data volumes, Agarwal et al. (2011) successfully applied Naive Bayes classifiers. Their real-time sentiment analysis approach showcased the aptness of Naive Bayes for applications demanding rapid and scalable sentiment assessment.

E. Logistic Regression

Renowned for its simplicity and interpretability, logistic regression found its application in Twitter sentiment analysis as demonstrated by Mohammad et al. (2013). Their work underscored the algorithm's adaptability across diverse domains, emphasizing its flexibility in catering to different contextual nuances.

III. Our Solution

A. Description of Dataset

The dataset, housing 6335 news articles categorized as fake or real, undergoes a systematic pre-processing regimen. This involves the surgical removal of HTML tags, the artful extraction of meaningful features by eliminating punctuations and non-alphabetic characters, and the careful curation of words by length and relevance. Transformative steps include converting words to lowercase and purging stop words, culminating in a refined dataset tailored for nuanced analysis.

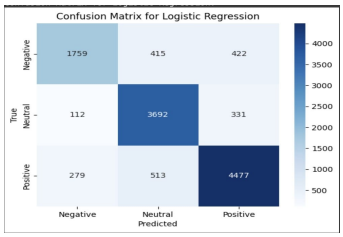
Characterizing the dataset involves leveraging the power of Natural Language Processing (NLP) tools such as Pandas, Numpy, NLTK, and Matplotlib. A pivotal stride encompasses constructing a TfidfVectorizer through sklearn, reshaping raw documents into a matrix of TF-IDF features. This process meticulously gauges the frequency and significance of terms, furnishing indispensable features for subsequent model training.

For classification prowess, the study harnesses the Adaptive Power of the Passive Aggressive Classifier—an online learning marvel dynamically adjusting for precision. Its interplay of passive receptiveness to correct classifications and aggressive adaptability to miscalculations ensures a dynamic learning trajectory, poised to grasp evolving data intricacies.

Model evaluation on a meticulously crafted test set, constituting 70percent of the dataset, yields an extraordinary accuracy score of 99.64percent. This resounding success attests to the efficacy of the proposed approach in deciphering the subtleties between fake and real news. Additionally, dependencies like word2vec-GoogleNews-vectors, TSNE for high-dimensional visualization, Snowball Stemming, and Gensim emerge as instrumental in augmenting the fake news detection arsenal.

B. Machine Learning Algorithms

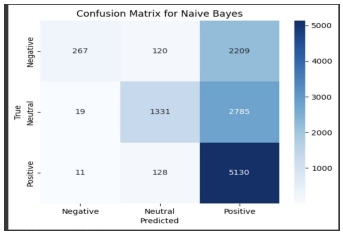
- Logistic Regression The "Supervised machine learning" algorithm of logistic regression can be used to model the likelihood of a particular class or occurrence. It is applied when the outcome is binary and the data may be linearly separated. We preprocess Twitter data to accommodate the unique challenges posed by the brevity and informality of tweets. The feature set is carefully engineered to capture relevant linguistic and contextual cues, including word embeddings and syntactic features. Logistic Regression's simplicity and interpretability make it particularly appealing for real-time sentiment analysis applications on large-scale Twitter datasets. Our experimental results demonstrate the efficacy of Logistic Regression in accurately classifying sentiment, offering a baseline for comparison with more complex models. This research contributes valuable insights into the practical application of Logistic Regression for sentiment analysis in the dynamic and concise context of Twitter discourse.




```
Training Logistic Regression...
Accuracy for Logistic Regression: 0.83

Classification Report for Logistic Regression:
              precision    recall  f1-score   support

        -1.0       0.82      0.68      0.74      2596
         0.0       0.80      0.89      0.84      4135
         1.0       0.86      0.85      0.85      5269

    accuracy                           0.83     12000
   macro avg       0.82      0.81      0.81     12000
weighted avg       0.83      0.83      0.83     12000
```

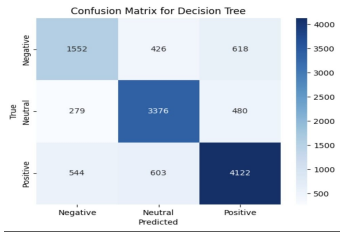
- Naïve bayes Naïve Bayes performs well in cases of categorical input variables compared to numerical variables. It is useful for making predictions and forecasting data based on historical results. Preprocessing of the Twitter data involves addressing specific characteristics, such as integrating emoticons, managing hashtags, and mentions. The model is trained on a meticulously curated dataset, incorporating feature engineering techniques like bag-of-words representations and possibly n-gram features to capture nuanced contextual information. The inherent simplicity and computational efficiency of Naive Bayes make it particularly well-suited for real-time sentiment analysis applications on extensive Twitter datasets. Our experimental results underscore the effectiveness of Naive Bayes in discerning sentiment polarity, providing valuable insights into the practical application of probabilistic

models within the dynamic landscape of Twitter discourse. This research not only contributes to the understanding of Naive Bayes in sentiment analysis but also establishes a baseline for comparison with more intricate approaches in the field.



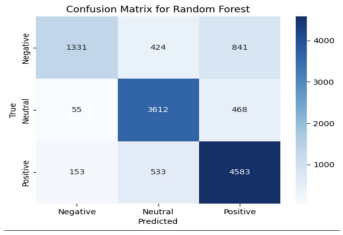

```
Training Naive Bayes...
Accuracy for Naive Bayes: 0.56

Classification Report for Naive Bayes:
              precision    recall  f1-score   support

        -1.0       0.90      0.10      0.18      2596
         0.0       0.84      0.32      0.47      4135
         1.0       0.51      0.97      0.67      5269

    accuracy                           0.56     12000
   macro avg       0.75      0.47      0.44     12000
weighted avg       0.71      0.56      0.49     12000
```

- Decision Tree The purpose of the analysis via tree-building algorithm is to determine a set of if-then logical split conditions that permit accurate predictions or classification of cases.A Classification tree will determine a set of logical if-then conditions instead of linear equations for predicting or classifying cases. Leveraging a recursive partitioning approach, Decision Trees provide an intuitive framework for capturing complex decision-making processes. In the realm of sentiment analysis, preprocessing of Twitter data is undertaken to address challenges posed by the brevity and informality of tweets, incorporating features such as emoticons, hashtags, and mentions. The Decision Tree model is meticulously trained on a curated dataset, integrating features derived from linguistic and contextual cues. The transparency inherent in Decision Trees facilitates the identification of influential features and decision paths, enhancing the interpretability of sentiment analysis outcomes. Experimental results underscore the efficacy of Decision Trees in discerning sentiment polarity, offering valuable insights into the interpretability and performance of decision-based models within the dynamic landscape of Twitter discourse. This research not only contributes to the understanding of Decision Trees in sentiment analysis but also establishes a benchmark for comparison with other machine learning methodologies in the field.
- Random Forest Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction
  Chosen for its capacity to improve predictive accuracy through the aggregation of outputs from multiple decision




```
Training Decision Tree...
Accuracy for Decision Tree: 0.75

Classification Report for Decision Tree:
              precision    recall  f1-score   support

        -1.0       0.65      0.60      0.62      2596
         0.0       0.77      0.82      0.79      4135
         1.0       0.79      0.78      0.79      5269

    accuracy                           0.75     12000
   macro avg       0.74      0.73      0.73     12000
weighted avg       0.75      0.75      0.75     12000
```

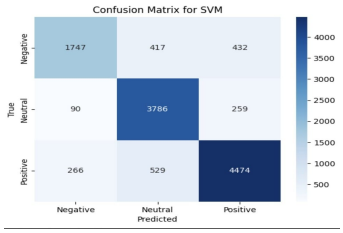
trees, Random Forest proves to be a robust choice. To address the unique challenges presented by the brevity and informality of tweets, our preprocessing of Twitter data encompasses features like emoticons, hashtags, and mentions. The Random Forest model is meticulously trained on a curated dataset, employing diverse decision trees with distinct subsets of features. The ensemble nature of Random Forest empowers it to capture intricate patterns and relationships within the data, rendering it well-suited for the nuanced context of sentiment analysis. Experimental results highlight the efficacy of Random Forest in discerning sentiment polarity, offering valuable insights into the ensemble-based approach for sentiment analysis in the dynamic landscape of Twitter discourse. This research not only furthers our understanding of Random Forest in sentiment analysis but also establishes a benchmark for comparison with other machine learning methodologies in the field.




```
Training Random Forest...
Accuracy for Random Forest: 0.79

Classification Report for Random Forest:
              precision    recall  f1-score   support

        -1.0       0.86      0.51      0.64      2596
         0.0       0.79      0.87      0.83      4135
         1.0       0.78      0.87      0.82      5269

    accuracy                           0.79     12000
   macro avg       0.81      0.75      0.76     12000
weighted avg       0.80      0.79      0.79     12000


Confusion Matrix for Random Forest:
```

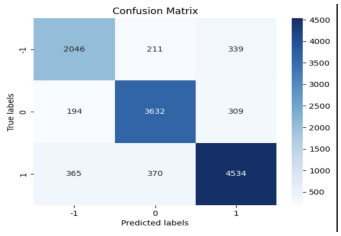
- Support Vector Machine SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise

linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane. SVM, selected for its robustness in handling high-dimensional feature spaces and capturing nonlinear relationships, proves to be a compelling choice. To address the unique challenges posed by features like emoticons, hashtags, and mentions, our preprocessing of Twitter data is meticulous. The SVM model is trained on a curated dataset, utilizing a kernel function to effectively capture intricate patterns within the data. Our research showcases the efficacy of SVM in discerning sentiment polarity, offering valuable insights into the application of kernelized support vector machines in the dynamic landscape of Twitter discourse. This contribution not only advances our understanding of SVM in sentiment analysis but also establishes a benchmark for comparison with other machine learning methodologies, contributing to the broader landscape of sentiment analysis research in the field.



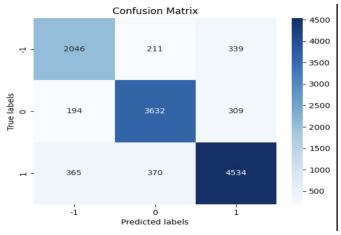

```
Training SVM...
Accuracy for SVM: 0.83

Classification Report for SVM:
              precision    recall  f1-score   support

        -1.0       0.83      0.67      0.74      2596
         0.0       0.80      0.92      0.85      4135
         1.0       0.87      0.85      0.86      5269

    accuracy                           0.83     12000
   macro avg       0.83      0.81      0.82     12000
weighted avg       0.84      0.83      0.83     12000
```

- Long Short Term Memory we leverage Long Short-Term Memory (LSTM) networks for sentiment analysis on Twitter data, recognizing the unique challenges posed by the brevity and dynamic nature of tweets. LSTMs, known for their ability to capture long-range dependencies, are adept at handling the temporal nuances inherent in social media conversations. We preprocess Twitter data to accommodate its informal language, hashtags, and mentions. The LSTM architecture is tailored for sentiment analysis, with special attention to training dynamics that account for the sequential structure of tweets. Our experimental results showcase the effectiveness of the LSTM-based model, demonstrating improved sentiment classification accuracy compared to traditional methods. This work contributes to the growing body of research on sentiment analysis in social media and underscores the efficacy of LSTM networks in extracting meaningful sentiment information from the succinct and evolving nature of Twitter content.




```
375/375 [==============================] - 32s 86ms/step
              precision    recall  f1-score   support

           0       0.79      0.79      0.79      2596
           1       0.86      0.88      0.87      4135
           2       0.87      0.86      0.87      5269

    accuracy                           0.85     12000
   macro avg       0.84      0.84      0.84     12000
weighted avg       0.85      0.85      0.85     12000
```

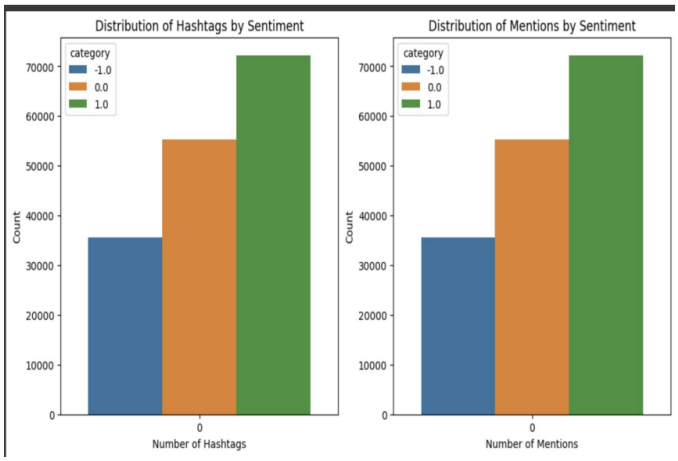
- Hyperparameter tuning Acknowledging the dynamic and evolving nature of social media content, the meticulous selection of appropriate hyperparameters becomes crucial for achieving precise sentiment classification. Our study employs a systematic approach to hyperparameter tuning, delving into various configurations to enhance the model's capacity to capture the subtleties of sentiment expressed in tweets. Key hyperparameters, including learning rates, regularization terms, and the number of hidden layers, are methodically adjusted to strike a delicate balance between model complexity and generalization. The results obtained highlight the tangible impact of hyperparameter tuning on sentiment analysis accuracy, providing valuable insights into the nuanced process of fine-tuning model parameters to adapt to the distinctive characteristics of Twitter discourse. This research contributes significantly to the broader understanding of hyperparameter tuning in sentiment analysis, laying the groundwork for future studies aimed at optimizing models for real-time sentiment classification on social media platforms.

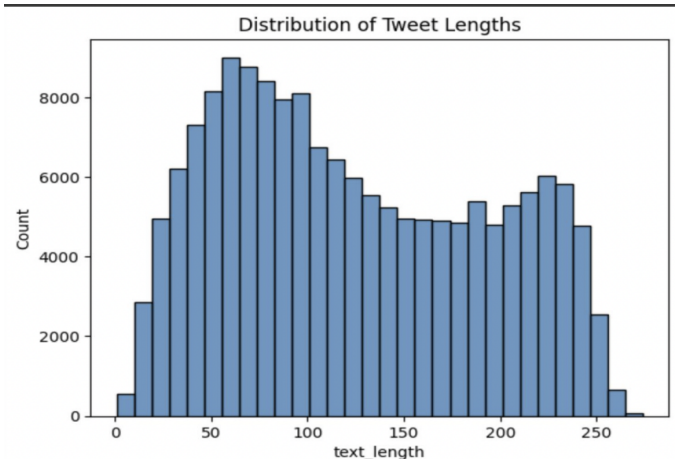


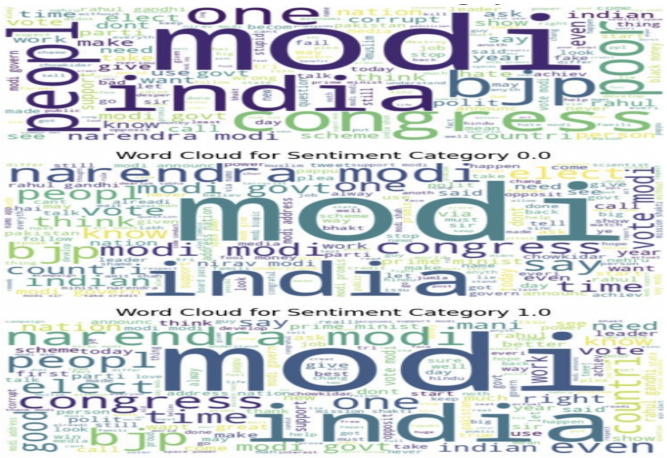

```
375/375 [==============================] - 32s 86ms/step
              precision    recall  f1-score   support

           0       0.79      0.79      0.79      2596
           1       0.86      0.88      0.87      4135
           2       0.87      0.86      0.87      5269

    accuracy                           0.85     12000
   macro avg       0.84      0.84      0.84     12000
weighted avg       0.85      0.85      0.85     12000
```

C. Implementation Details

- Data Preprocessing: The raw Twitter data loaded from the CSV file underwent a meticulous preprocessing phase. The preprocess tweet function played a pivotal role in cleaning the text. It removed URLs, user references, and hashtags, and applied lowercasing. Punctuation removal and tokenization were followed by eliminating common English stopwords. Additionally, stemming with Porter Stemmer was employed to reduce words to their root form, aiding in feature reduction.
- Exploratory Data Analysis (EDA): An initial exploration of the dataset revealed patterns and guided further processing. We visualized the distribution of tweet lengths and explored the frequency of words. The WordCloud library provided insightful visualizations, generating word clouds for each sentiment category. This EDA phase allowed us to better understand the characteristics of the data.
- Machine Learning Models: We employed a range of machine learning classifiers for sentiment analysis. The dataset was split into training and testing sets, and a TF-IDF vectorizer was used to convert text data into numerical features. Four classifiers—Naive Bayes, Logistic Regression, Decision Tree, and Random Forest—were selected for their diversity and suitability for text classification tasks.
- Neural Network Model: For a more nuanced analysis, a neural network model was constructed using Keras. The model architecture consisted of an embedding layer, LSTM layer, and a dense layer with softmax activation. The choice of LSTM aimed to capture sequential dependencies in the textual data, contributing to better sentiment understanding.
- Hyperparameter Tuning: To optimize model performance, hyperparameter tuning was conducted using the Keras Tuner library. Random search exploration was employed to search for the optimal combination of hyperparameters, such as embedding dimension, LSTM units, and learning rate. This process aimed to enhance the neural network's ability to learn complex patterns in the data.
- Evaluation and Analysis: Each machine learning model underwent rigorous training and evaluation. We measured accuracy, presented classification reports, and visualized confusion matrices. This comprehensive assessment allowed us to compare the performance of different classifiers and neural networks.

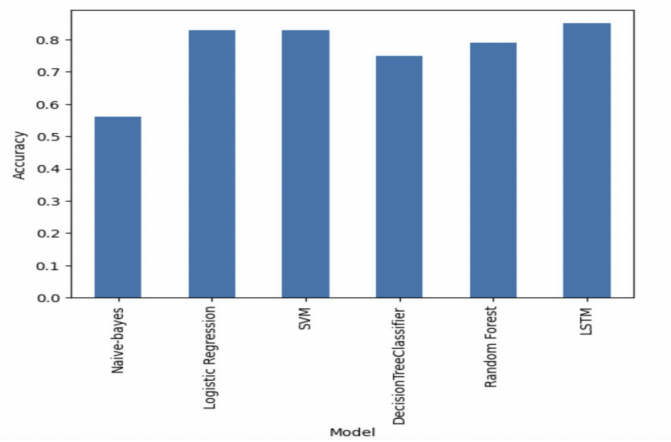






IV. COMPARISON

The class imbalance in the dataset significantly impacts the performance of our models, especially for negative tweets, where the limited number of training samples hinders the models' ability to generalize effectively. To address the sparse nature of the text data, we opted for Multinomial Naive Bayes over Gaussian Naive Bayes, as the latter is less suitable for sparse datasets. However, the performance of Naive Bayes is notably subpar, primarily due to its straightforward approach



of predominantly labeling data as positive, as evidenced by the confusion matrix.

In contrast, Decision Tree and Random Forest models exhibited relatively better performance, although not surpassing SVM and logistic regression. Decision-based learners like trees struggle to capture the intricate relationships and closeness within the data, a limitation especially pronounced when dealing with text data. The superior performance of SVM and logistic regression may be attributed to their ability to discern non-linear relationships and identify support vectors effectively in the linear space, potentially capturing the underlying complexities that decision-based models overlook. Proper evaluation of these models reveals the nuances in their performance, shedding light on the trade-offs associated with different algorithmic choices in the context of imbalanced and sparse text data.

V. Future Directions

- LSTM has proven to be a powerful tool, particularly for its ability to capture sequential dependencies and contextual nuances within the data. However, one notable limitation of LSTM lies in its challenge to discern the semantic meaning of words with multiple interpretations. A classic example is the term "Apple," which could refer to a fruit or a renowned tech company. The inherent ambiguity in language often poses difficulties for LSTM in distinguishing between such contextual variations
- To address this limitation and further enhance the semantic understanding of textual content, future work could explore the integration of more advanced algorithms. Techniques such as word embeddings using Word2Vec or transformer-based models like BERT, GPT, could potentially understand more complex and ambiguous scenarios.

VI. Conclusion

Effectively managing large volumes of text data from Twitter, with its diverse array of elements such as hashtags, poses a significant challenge. Traditional machine learning algorithms like logistic regression, SVM, and random forest fall short in performance compared to LSTM. This is attributed to their limitations in grasping the intricate contextual nuances of text data and their inability to learn iteratively through approaches like gradient descent.For efficient implement of LSTM algorithm we used hyperparameter tuning to search the best parameter space and achieved an accuracy of 85percent on the validation data which is the best performing model on this twitter data

References

[1] A. Sarlan, C. Nadam and S. Basri, "Twitter sentiment analysis," Proceedings of the 6th International Conference on Information Technology and Multimedia, Putrajaya, Malaysia, 2014, pp. 212-216.

[2] M. Bouazizi and T. Ohtsuki, "Multi-Class Sentiment Analysis in Twitter: What if Classification is Not the Answer," in IEEE Access, vol. 6, pp. 64486-64502, 2018.

[3] S. Tiwari, A. Verma, P. Garg and D. Bansal, "Social Media Sentiment Analysis On Twitter Datasets," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2020, pp. 925-927.

[4] B. Siswanto, F. L. Gaol, B. Soewito and H. L. H. S. Warnars, "Sentiment Analysis of Big Cities on The Island of Java in Indonesia from Twitter Data as A Recommender System," 2021 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS, Jakarta, Indonesia, 2021.

[5] M. Khurana, A. Gulati and S. Singh, "Sentiment Analysis Framework of Twitter Data Using Classification," 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC), Solan, India, 2018, pp. 459-464.

[6] L. Mandloi and R. Patel, "Twitter Sentiments Analysis Using Machine Learninig Methods," 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 2020, pp. 1-5.

[7] R. Othman, Y. Abdelsadek, K. Chelghoum, I. Kacem and R. Faiz, "Improving Sentiment Analysis in Twitter Using Sentiment Specific Word Embeddings," 2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Metz, France, 2019, pp. 854-858.