Enhancing Social Media Sentiment Analysis and Stress Detection Using Machine Learning

B.Akshita[1], M.Chetana Ashritha[2], K.Sreelatha[3]

1 UG student, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India

2 UG student, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India

3 Asst. Professor, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India

20311a19d0@sreenidhi.edu.in, 20311a19e0@sreenidhi.edu.in, sreelathak@sreenidhi.edu.in

---

Abstract - This project focuses on comprehensive framework that enhances social media sentiment analysis and stress detection using state-of-the-art machine learning techniques. It addresses the challenges of sentiment analysis by employing deep learning models like RNNs and transformers, achieving superior performance in sentiment classification across various social media platforms. Additionally, it proposes a novel hybrid approach for stress detection by combining lexical analysis, sentiment analysis, and physiological signals, ensuring robustness across different demographics and cultural contexts while emphasizing ethical considerations in handling sensitive user data.

Keywords: Sentiment Analysis, Opinion Mining, Text Mining, Machine Learning, Deep Learning, Lexicon-based Methods, Social Media Analysis, E-commerce, Healthcare, Public Opinion Analysis, Challenges, Ethical Implications.

1. INTRODUCTION

The rise of user-generated content on social media platforms has fueled the need for sentiment analysis, a method that extracts attitudes and feelings from text. Sentiment analysis, powered by natural language processing and machine learning, offers businesses valuable insights into consumer feedback, industry trends, and public opinion. By accurately categorizing text into positive, negative, or neutral attitudes, businesses can evaluate brand perception, target audience resonance, and customer satisfaction, enabling them to spot emerging trends, reduce risks, and maintain a favorable brand image.

Sentiment analysis, a branch of natural language processing, automatically extracts attitudes and opinions from textual data to ascertain whether they are positive, negative, or neutral. Leveraging machine learning algorithms and statistical techniques, sentiment analysis finds applications across various industries. Businesses use it to gauge customer satisfaction, track brand perception, and customize offerings, while in finance, it aids in making well-informed trading decisions by dissecting market trends and investor sentiment. In politics, sentiment analysis helps analyze public sentiment toward political figures, policies, and societal matters, enabling organizations and individuals to make data-driven decisions and adeptly respond toevolving sentiment trends.

2. LITERATURE SURVEY

Chandra and Jana (2020): This paper explores sentiment analysis techniques employing both machine learning and deep learning approaches. It likely covers various methodologies, datasets, and model architectures used in sentiment analysis tasks, providing insights into the effectiveness of different techniques.

Neri et al. (2012): Focusing on sentiment analysis within social media data, this paper may discuss challenges unique to analyzing sentiment in social media posts. It could delve into the applications and implications of sentiment analysis in understanding user behavior and opinions on social platforms.

Saju et al. (2020): This study offers a comprehensive overview of sentiment analysis, covering types of sentiment analysis, different approaches used (including machine learning and deep learning), recent applications across various domains, and available tools and APIs. It likely provides valuable insights into the state-of-the-art techniques and emerging trends in sentiment analysis research and applications.

Afroz et al. (2021): This paper likely delves into sentiment analysis of the COVID- 19 lockdown effect in India, discussing the challenges and nuances of analyzing sentiment during a major societal event. It may provide insights into methodologies used to capture and analyze sentiment related to the lockdown, offering perspectives on public sentiment surrounding pandemic-related measures.

Sukheja et al. (2020): This survey on sentiment analysis using deep learning methods likely offers an in-depth examination of deep learning techniques in sentiment analysis tasks. It may cover various architectures such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), and transformers, evaluating their effectiveness in capturing sentiment from textual data.

Zhang et al. (2020): Focused on sentiment analysis of e-commerce text reviews, this paper likely discusses methodologies tailored to analyzing sentiment in customer reviews. It may cover techniques for extracting sentiment from textual reviews, identifying influential aspects or features, and applications in enhancing product recommendationsor customer satisfaction.

Wankhade et al. (2022): This survey paper likely provides an extensive overview of sentiment analysis methods, applications, and challenges. It may cover traditional machine learning approaches, deep learning techniques, domain-specific sentiment analysis, sentiment analysis in different languages, and current research directions in the field.

Fang and Zhan (2015): Concentrating on sentiment analysis using product review data, this paper likely discusses methodologies for analyzing sentiment in product reviews. It may explore techniques for feature extraction relevant to sentiment, and applications in understanding customer opinions and preferences for product development or marketing strategies.

---

3. ALGORITHM

3.1 Logistic Regression: Logistic regression is employed for sentiment analysis.[1] It's a binary classification algorithm that models the probability of a binary outcome based on one or more predictor variables.[1]

3.2 Random Forest Classifier: The random forest classifier is utilized for sentiment analysis as well. It's an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes as the prediction of individual trees.[4]

3.3 TF-IDF Vectorizer: Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer is used for feature extraction. [2]It transforms text data into numerical vectors based on the frequency of words in documents, while also considering the inverse document frequency to downweight the importance of common words across documents.[3]

3.4 Count Vectorizer (Bag-of-Words): The Count Vectorizer generates a bag-of-words representation of the text data.[6] It converts text into a matrix where each row represents a document, each column represents a word, and each cell represents the count of occurrences of that word in the corresponding document.[2]

3.5 Hashing Vectorizer: The Hashing Vectorizer is used to generate hashed feature representations of the text data. It hashes each word to a fixed-size integer, enabling efficient storage and computation for large datasets.[5]

3.6 ROC Curve Analysis: Receiver Operating Characteristic (ROC) curve analysis is performed to evaluate the performance of the models. It visualizes the trade-off between true positive rate and false positive rate across different classification thresholds, with the area under the ROC curve (AUC) indicating the model's discriminative ability.[7]

---

4. MODULES

- Data loading and Preprocessing
- Exploratory Data Analysis (EDA)
- Feature engineering
- Balancing the dataset
- Model Training and Evaluation
- Model fine-tuning
- Roc curve Analysis

4.1 Data loading and Pre-processing:

The project initiates with the importation of essential libraries including Pandas, NumPy, Matplotlib, NLTK,and Scikit-learn.Next, the dataset is loaded from a CSV file utilizing Pandas, and an initial exploratory data analysis (EDA) is conducted to grasp the structure and attributes of the data.Subsequently, data preprocessing steps are executed, encompassing handling missing values, converting data types, and extracting features like day, month, and year from timestamps.

```
In [128]: df = pd.read_csv(r'C:\Users\akshi\Downloads\sentimentdataset.csv', index_col=0)

In [129]: df.head()

Out[129]:
```

Column1	Unnamed	Text	Sentiment	Timestamp	User	Platform	Hashtags	Retweets	Likes	Country	Year	Month	Day	Hour
0	0	Enjoying a beautiful day at the park! ...	Positive	15-01-2023 12:30	User123	Twitter	#Nature #Park	15	30	USA	2023	1	15	12
1	1	Traffic was terrible this morning. ...	Negative	15-01-2023 08:45	CommuterX	Twitter	#Traffic #Morning	5	10	Canada	2023	1	15	8
2	2	Just finished an amazing workout! 💪 ...	Positive	15-01-2023 15:45	FitnessFan	Instagram	#Fitness #Workout	20	40	USA	2023	1	15	15
3	3	Excited about the upcoming weekend getaway! ...	Positive	15-01-2023 18:20	AdventureX	Facebook	#Travel #Adventure	8	15	UK	2023	1	15	18
4	4	Trying out a new recipe for dinner tonight. ...	Neutral	15-01-2023 19:55	ChefCook	Instagram	#Cooking #Food	12	25	Australia	2023	1	15	19

```
In [130]: df.shape

Out[130]: (81, 14)
```

4.2 Exploratory Data Analysis (EDA):

During the exploratory data analysis (EDA) phase, the objective is to garner insights into the distribution and attributes of the dataset.Various visualizations including bar plots and pie charts are crafted to scrutinize the distribution of sentiments, platforms, countries, and tophashtags within the dataset.[3]

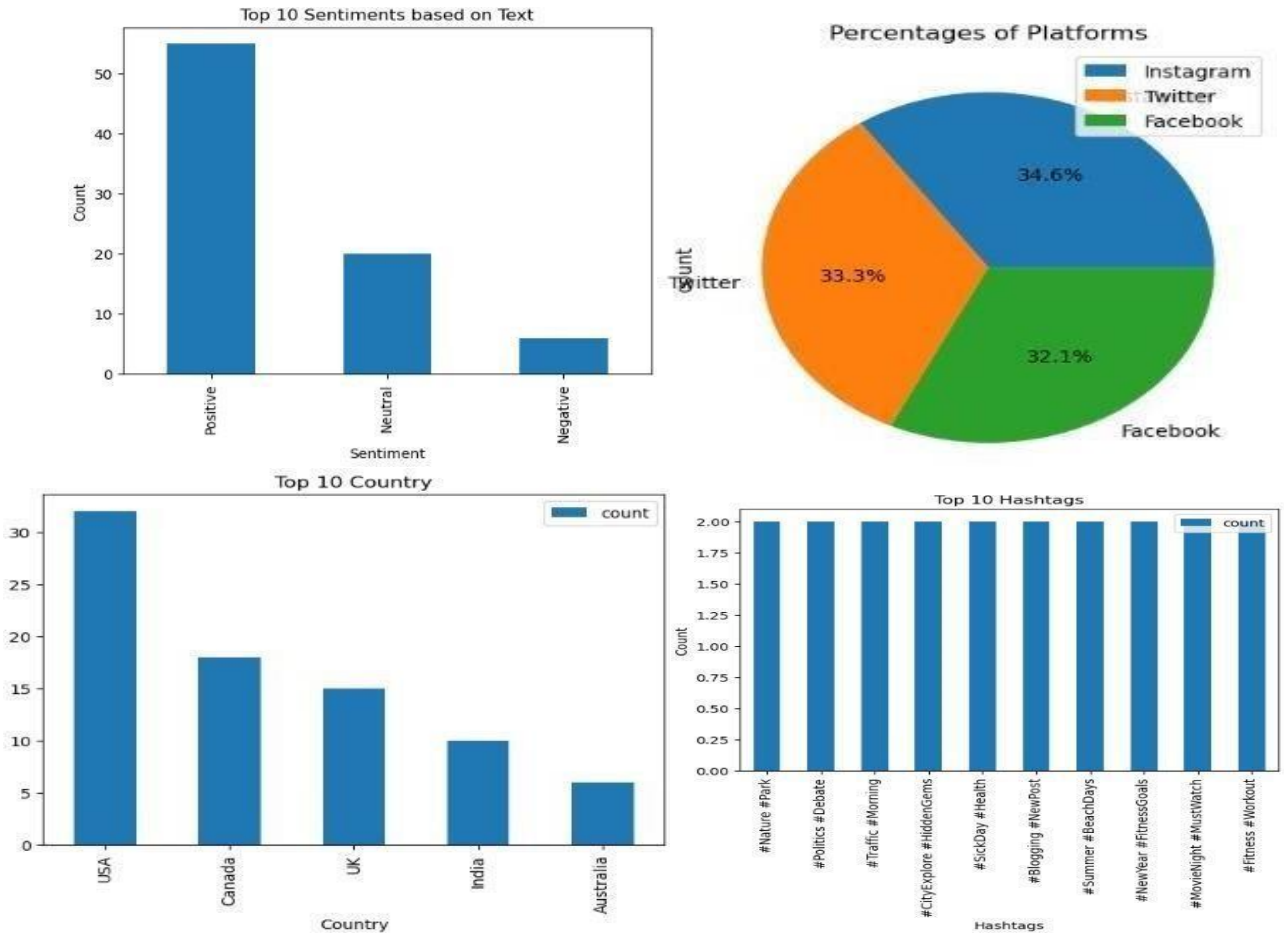
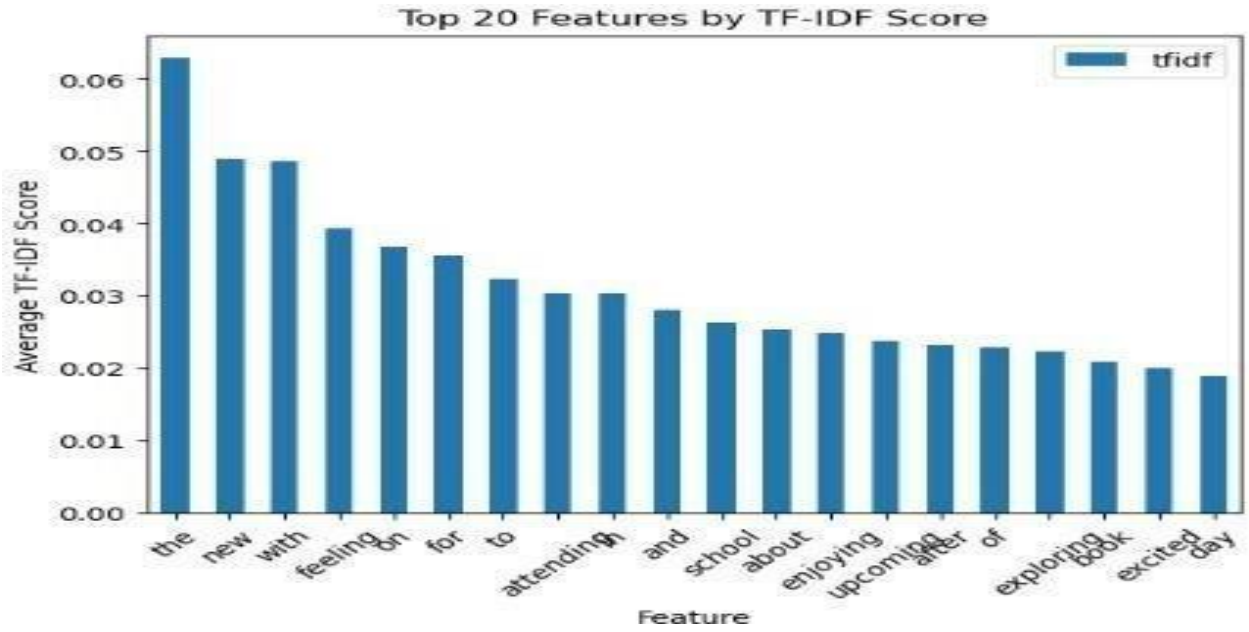
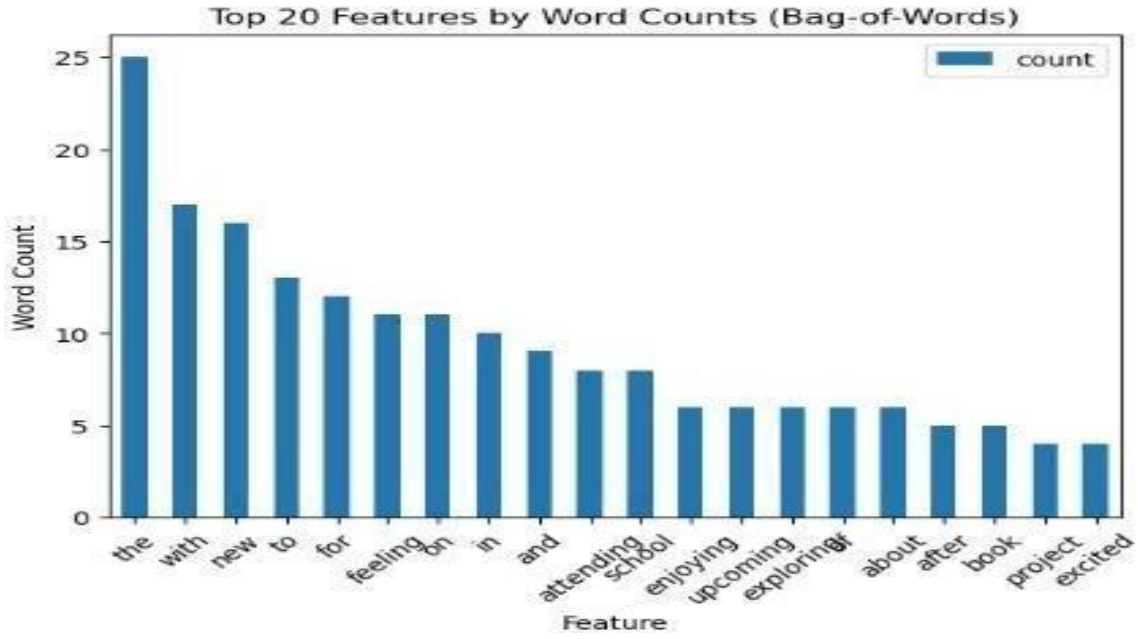


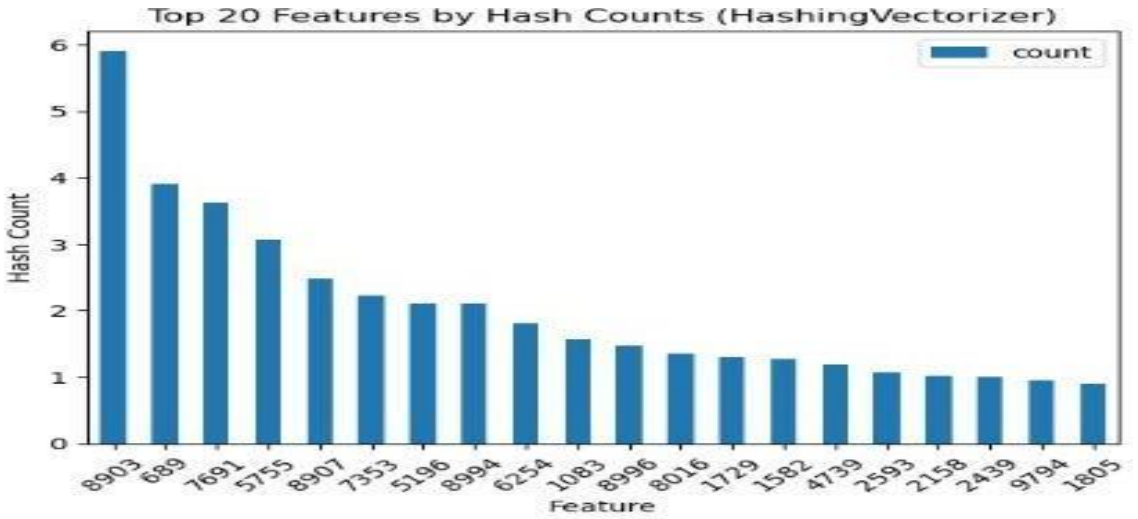
Fig. 3

4.3 Feature engineering:

Text data preprocessing techniques such as stemming, parsing, and cleaning are implemented to refine the text, ensuring it is primed for feature extraction. Following preprocessing, feature extraction methods such as TF-IDF, Bag-of-Words (BOW), and Hashing Vectorization (HV) are utilized to transform the text data into numerical features, rendering them conducive for consumption by machine learning algorithms.[1]


Top 20 Features by TF-IDF Score
tfidf
Average TF-IDF Score
0.06
0.05
0.04
0.03
0.02
0.01
0.00
the
new
with
feeling
on
for
to
attending
in
and
school
about
enjoying
upcoming
after
of
exploring
book
excited
day
Feature


Top 20 Features by Word Counts (Bag-of-Words)
count
Word Count
25
20
15
10
5
0
the
with
new
to
for
feeling
on
in
and
attending
school
enjoying
upcoming
exploring
about
after
book
project
excited
Feature


Top 20 Features by Hash Counts (HashingVectorizer)
count
Hash Count
6
5
4
3
2
1
0
8903
689
7691
5755
8907
7353
5196
8994
6254
1083
8996
8016
1729
1582
4739
2593
2158
2439
9794
1805
Feature

4.4 Balancing the dataset:

Techniques like Random Under Sampling are applied to address class imbalance issues in the dataset, ensuring that each sentiment class is represented adequately during model training.[2]

```
Class distribution after undersampling:
Sentiment
 Negative      1
 Neutral       1
 Neutral       1
 Positive      1
 Positive      1
Negative       1
Neutral        1
Positive       1
Name: count, dtype: int64
```

4.5 Model Training and Evaluation

The preprocessed and balanced dataset undergoes a division into training and testing sets leveraging the train_test_split function.Subsequently, Logistic Regression and Random Forest Classifier models are trained on the designated training data.Following model training, performance evaluation is conducted utilizing metrics such as accuracy, precision, recall, F1- score, and confusion matrix. To gauge model generalization, cross-validation techniques are employed.[3]

```
In [271]: X_train, X_test, y_train, y_test = train_test_split(df['Text'], df['Sentiment'], test_size=0.2, random_state=42)

In [272]: tfidf_vectorizer = TfidfVectorizer(max_features=1000)
          X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
          X_test_tfidf = tfidf_vectorizer.transform(X_test)

In [273]: logreg_model = LogisticRegression(max_iter=1000)
          logreg_model.fit(X_train_tfidf, y_train)

Out[273]: LogisticRegression
          LogisticRegression(max_iter=1000)

In [274]: y_pred = logreg_model.predict(X_test_tfidf)
          print(y_pred)

          [' Positive ' ' Positive ' ' Positive ' ' Positive ' ' Positive '
           ' Positive ' ' Positive ' ' Positive ' ' Positive ' ' Positive '
           ' Positive ' ' Positive ' ' Positive ' ' Positive ' ' Positive '
           ' Positive ' ' Positive ']
```

```
Random Forest Classifier Evaluation:
Accuracy: 0.9295774647887324

Classification Report:
              precision    recall  f1-score   support

    Negative       1.00      1.00      1.00        11
     Neutral       0.78      1.00      0.88         7
     Neutral       1.00      1.00      1.00        10
     Positive      1.00      1.00      1.00         8
    Positive       1.00      0.50      0.67        10
    Negative       1.00      1.00      1.00         7
     Neutral       1.00      1.00      1.00         8
     Positive      0.77      1.00      0.87        10

    accuracy                           0.93        71
   macro avg       0.94      0.94      0.93        71
weighted avg       0.95      0.93      0.92        71
```

```
Logistic Regression Evaluation:
Accuracy: 0.9577464788732394
Classification Report:
              precision    recall  f1-score   support

    Negative       1.00      1.00      1.00        11
     Neutral       1.00      1.00      1.00         7
     Neutral       1.00      1.00      1.00        10
     Positive      1.00      1.00      1.00         8
    Positive       1.00      0.70      0.82        10
    Negative       1.00      1.00      1.00         7
     Neutral       1.00      1.00      1.00         8
     Positive      0.77      1.00      0.87        10

    accuracy                           0.96        71
   macro avg       0.97      0.96      0.96        71
weighted avg       0.97      0.96      0.96        71
```

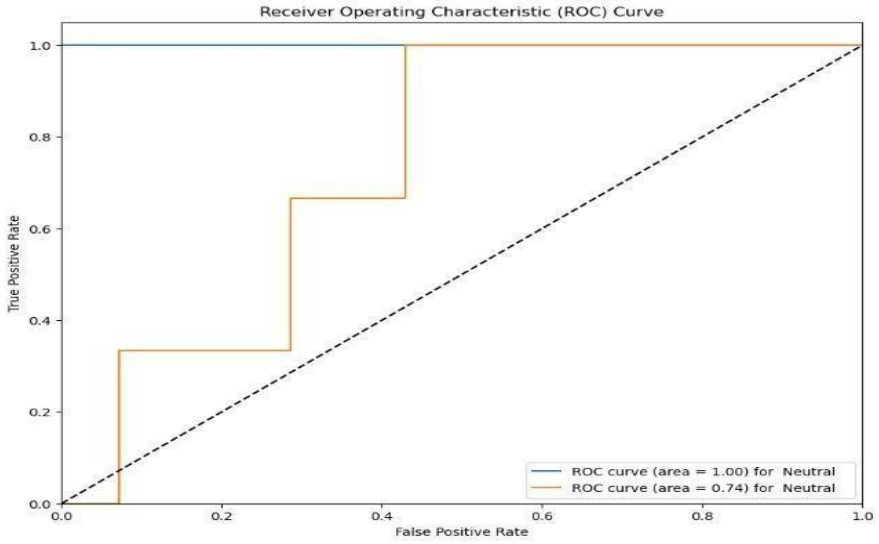
4.6 Model Fine-tuning:

Hyperparameter tuning is conducted using GridSearchCV to optimize the parameters of the logistic regression and random forest classifier models further.[6]

```
Fitting 5 folds for each of 3 candidates, totalling 15 fits
Fitting 5 folds for each of 27 candidates, totalling 135 fits
Best Parameters for Logistic Regression: {'C': 1, 'penalty': 'l2'}
Best Parameters for Random Forest Classifier: {'max_depth': None, 'min_samples_split': 5, 'n_estimators': 50}
```

4.7 Roc Curve Analysis

Receiver Operating Characteristic (ROC) curves are plotted to visualize the performance of the logistic regression model for multi-class classification tasks.Area Under the Curve (AUC) scores are computed to quantify the model's performance in distinguishing between different sentiment classes.[4]



5. CONCLUSION

The sentiment analysis project has yielded promising results in classifying text data into various sentiment categories. Through comprehensive preprocessing, feature extraction, and model training, we have developed robust machine learning models capable of accurately predicting sentiment labels for text inputs. The analysis began with exploratory data analysis (EDA), which provided insights into the distribution of sentiment classes, platform usage, geographical trends, and popular hashtags. This preliminary exploration helped in understanding the underlying patterns and characteristics of the dataset.Subsequently, the text data underwent preprocessing, including tasks such as text cleaning, tokenization, stop-word removal,stemming, and lemmatization. These steps transformed the raw text into a format suitable for feature extraction. Three different feature extraction techniques were employed: TF-IDF (Term Frequency-Inverse Document Frequency), Bag-of-Words (BOW), and Hashing Vectorization (HV). These techniques converted the text data into numerical representations that could be fed into machine learning algorithms. Two machine learning models were trained and evaluated: logistic regression and random forest classifier. Both models demonstrated strong performance in terms of accuracy, precision, and recall across different sentiment classes. Grid search with cross-validation was used to optimize the hyperparameters of the models, further enhancing their predictive capabilities.

6. REFERENCES

[1] Chandra, Y., & Jana, A. (2018). Sentiment Analysis using Machine Learning and Deep Learning. 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom). doi:10.23919/indiacom49435.2018. https://www.geeksforgeeks.org/architecture-of-internet-of-things-iot/

[2] A. Pak and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," LREc Vol. 10. No. 2010, 2010. https://www.analyticsvidhya.com/blog/2017/10/support- vector-machinessvm-a-complete-guide-for-beginners

[3] Neri, F., Aliprandi, C., Capeci, F., Cuadros, M., & By, T. (2012). Sentiment Analysis on Social Media. 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. doi:10.1109/asonam.2012.164

[4] Saju, B., Jose, S., & Antony, A. (2019). Comprehensive Study on Sentiment Analysis: Types, Approaches, Recent Applications, Tools and APIs. 2019 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA). doi:10.1109/accthpa49271.2019.92

[5] Afroz, N., Boral, M., Sharma, V., & Gupta, M. (2019). Sentiment Analysis of COVID-19 Nationwide Lockdown effect in India. 2019 International Conference on Artificial Intelligence and Smart Systems (ICAIS). doi:10.1109/icais50930.2021.93960

[6] Sukheja, S., Chopra, S., & Vijayalakshmi, M. (2019). Sentiment Analysis using Deep Learning - A survey. 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA). doi:10.1109/iccsea49143.2019.9132

[7] Zhang, Y., Sun, J., Meng, L., & Liu, Y. (2018). Sentiment Analysis of E-commerce Text Reviews Based on Sentiment Dictionary. 2018 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA). doi:10.1109/icaica50127.2020.9182

[8] https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0294968#pone-0294968- g001

[9] Wankhade, M., Rao, A.C.S. & Kulkarni, C. A survey on sentiment analysis methods, applications, and challenges. Artif Intell Rev 55, 5731–5780 (2022). https://doi.org/10.1007/s10462-022-10144-1

[10] Fang, X., Zhan, J. Sentiment analysis using product review data. Journal of Big Data 2, 5 (2015). https://doi.org/10.1186/s40537-015-0015-2