NLP Analysis of Crime Incident Reports

Introduction:

This report presents an NLP-based analysis aimed at supporting the development of a model to assist citizens in accurately filing cybercrime reports on the National Cyber Crime Reporting Portal (NCRP). Given the complex nature of cybercrime incidents, individuals often face challenges in articulating relevant details and categorizing incidents effectively. Through real-time analysis of report descriptions and supporting media files uploaded by users, this model seeks to streamline the reporting process by identifying sentiment trends, recognizing commonly reported cybercrime themes, and providing feedback to ensure accuracy in report submissions.

Our analysis leverages sentiment analysis to track the tone and urgency of incident reports, identifying patterns that can help classify the severity and nature of reported cases. Topic modelling further reveals frequently reported issues, such as phishing, identity theft, and financial fraud, which will guide users in describing incidents using relevant terminology.

Problem Statement:

The project addresses a **crime data classification problem**, where the goal is to categorize criminal activities based on textual descriptions (crimeaditionalinfo). The dataset includes multiple categories of crimes, and the task is to analyse, pre-process, and classify these activities accurately. The primary challenge lies in handling noisy and unstructured text data, performing effective feature extraction, and building a robust classification model capable of generalizing well on unseen data.

Implementation Steps:

1. **Data Loading and Cleaning**

    - Load datasets (train.csv and test.csv) and inspect their structure.
    - Remove duplicate and null rows to ensure data consistency.
    - Align categories in the training and test datasets for compatibility.

2. **Exploratory Data Analysis (EDA)**

    - Generate word frequency distributions and word clouds to understand text patterns.
    - Perform sentiment analysis to capture the emotional tone of the data.

3. **Text Pre-processing**

    - Tokenization, lemmatization, stop word removal, and text cleaning using spaCy.
    - Use TF-IDF Vectorizer to convert text data into numerical features.

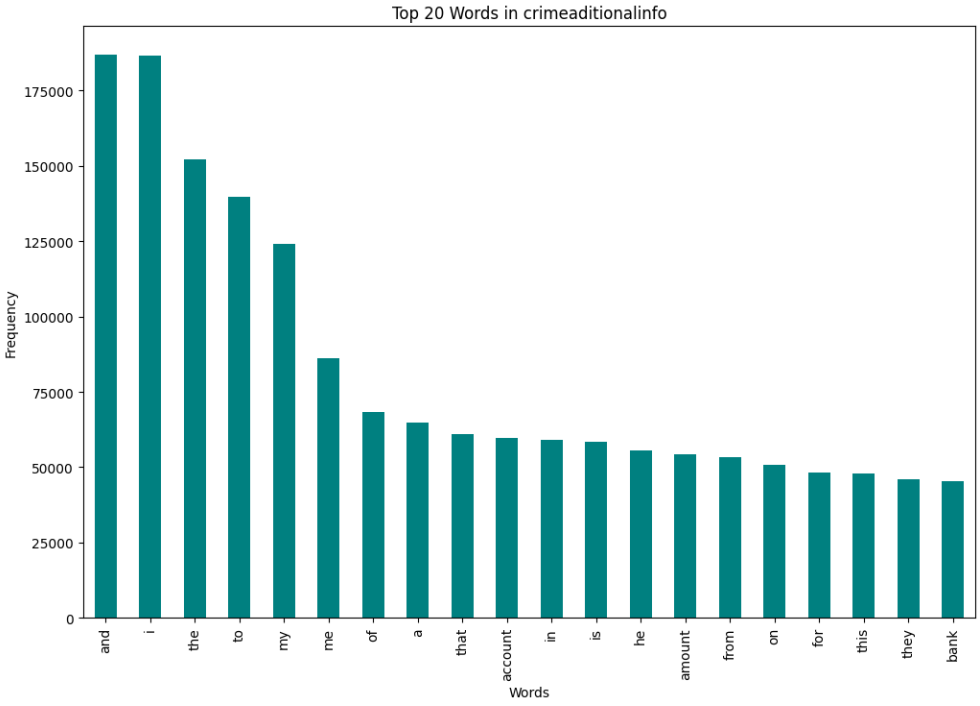
4. **Model Training and Evaluation**

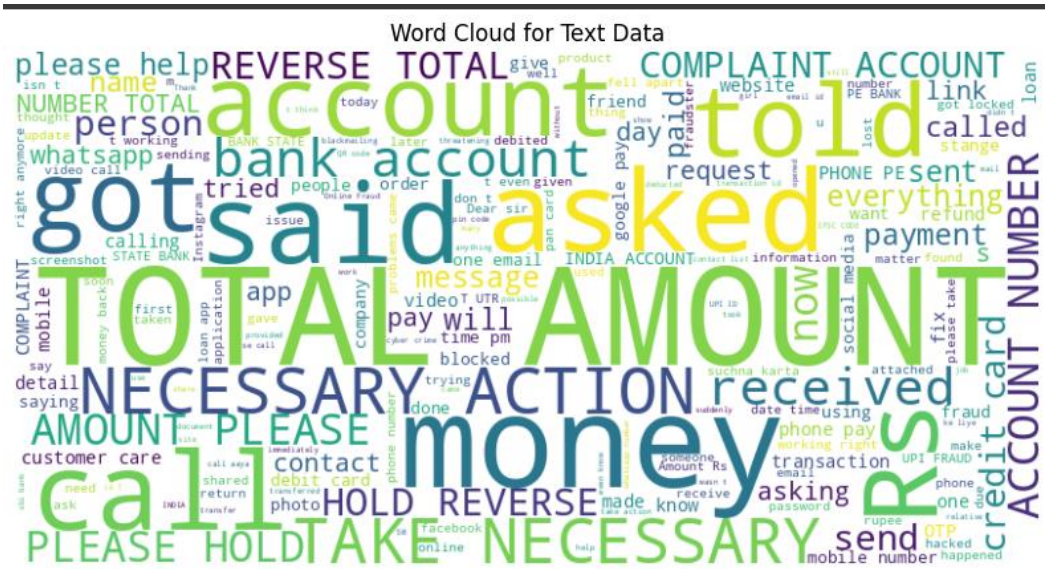
    - Train the Random Forest Classifier on the processed training data.
    - Evaluate the model using metrics such as accuracy, precision, recall, and F1-score.

Pre-processing Techniques:

- **Tokenization**: Splitting text into individual words or tokens for analysis.
- **Stop Word Removal**: Eliminating common words (e.g., "the," "and") that do not add value.
- **Lemmatization**: Reducing words to their root form to standardize text data.
- **Noise Removal**: Removing punctuations, numbers, and special characters for cleaner input.

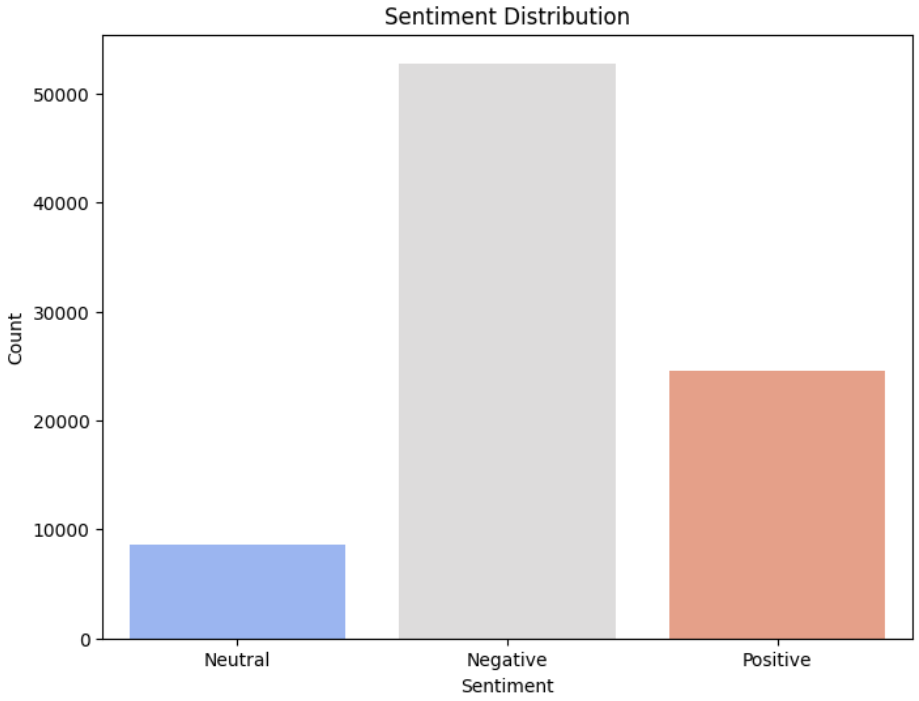
These techniques improve the quality of data representation, ensuring meaningful patterns are captured by the model.





Sentiment Analysis Findings:

The sentiment analysis of the dataset reveals an abundance of negative sentiment, which is consistent with the nature of the data, which primarily consists of crime reports. Negative feelings describe situations that have a negative impact on individuals or institutions, such as financial fraud or offensive content. Plotting these views over time reveals trends that allow the identification of periods with an upsurge in specific incidences. Peaks in negative sentiment may be indicative of illegal activity spikes or seasonal trends in specific sorts of crimes. Such insights are useful for resource allocation, as they enable organizations to organize actions around high-incidence intervals.



Model Implementation :

The **Random Forest Classifier** is chosen because:

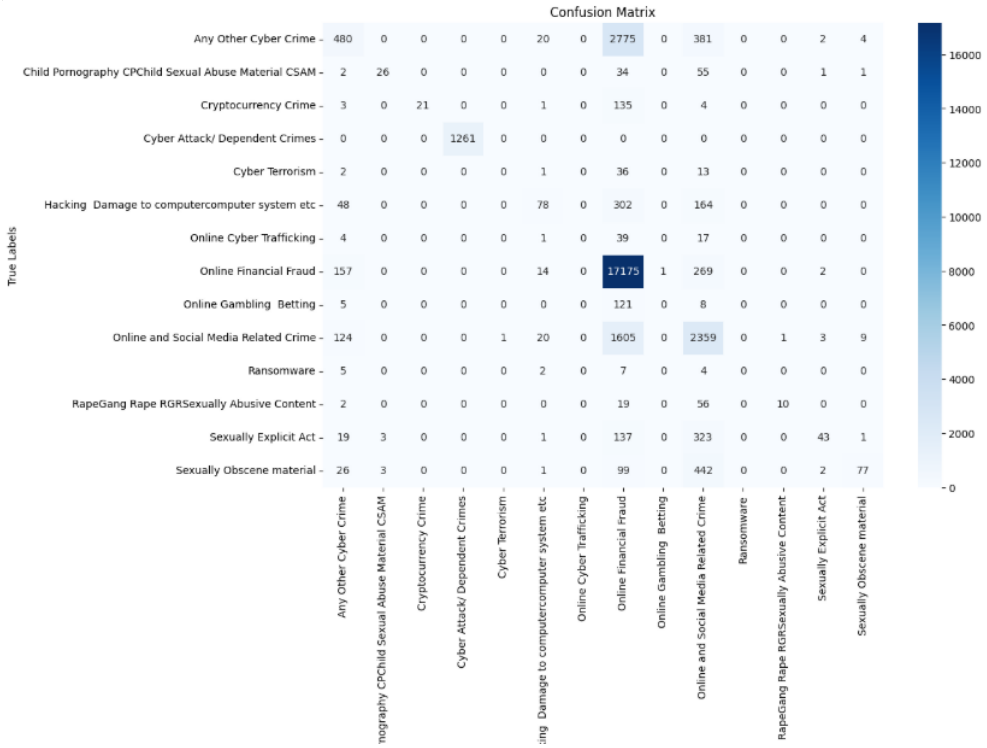
1. **Robustness**: It handles high-dimensional data and avoids overfitting due to its ensemble nature.
2. **Versatility**: Suitable for multi-class text classification tasks.
3. **Feature Importance**: Provides insights into the most critical features influencing predictions.

Working:

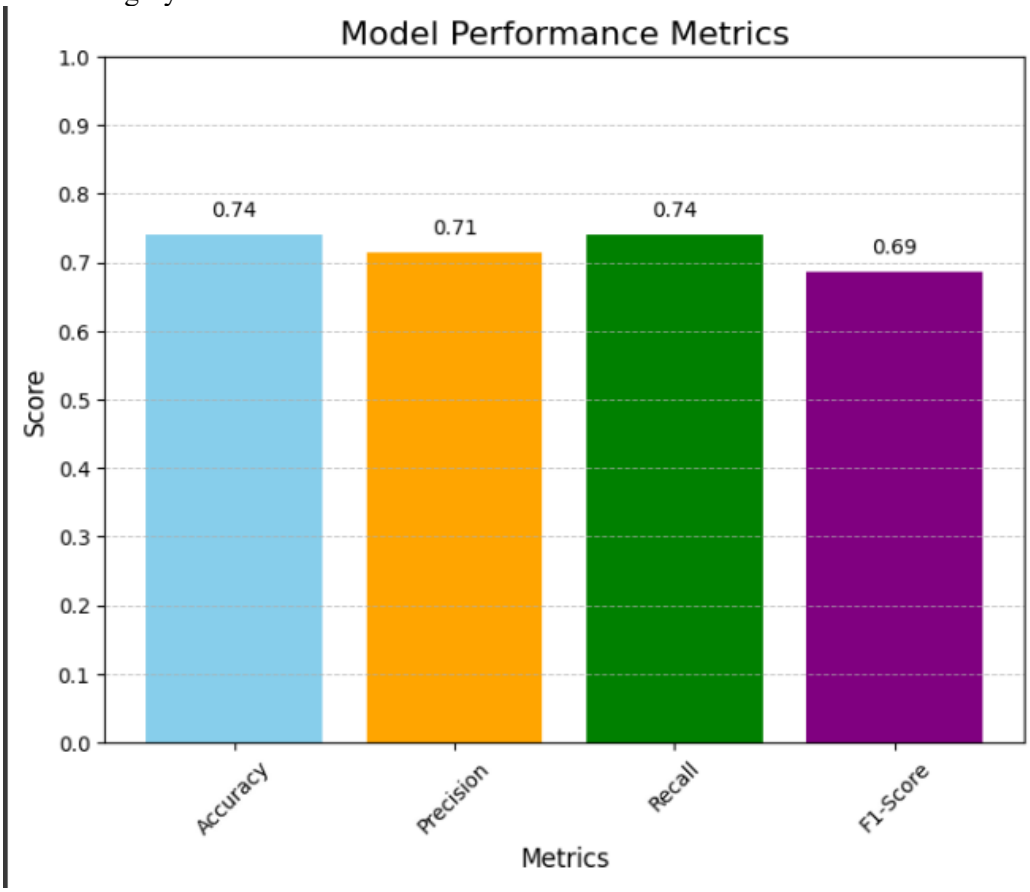
- The model builds multiple decision trees during training.
- Each tree predicts the class, and the final classification is based on the majority vote.
- This ensemble approach improves accuracy and reduces variance, making it a reliable choice for the dataset.

Visualization for Evaluation of the Model :

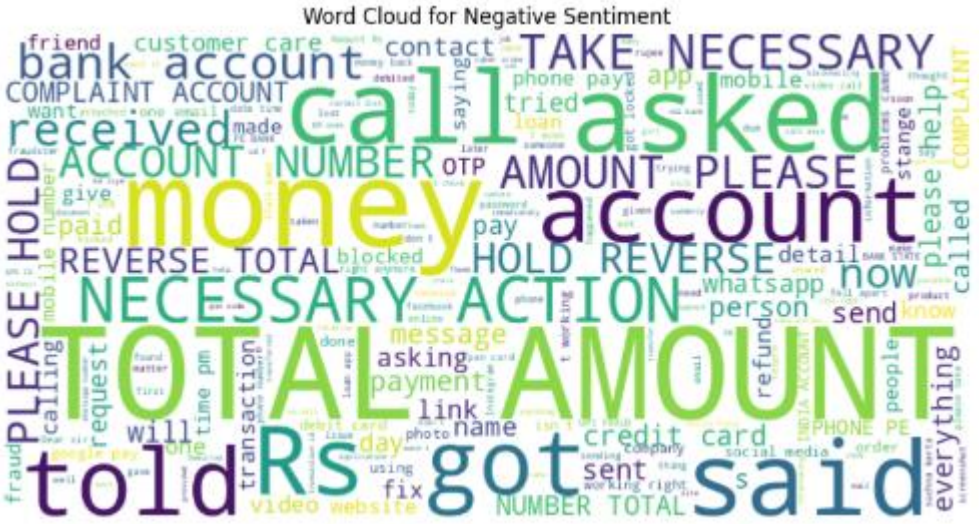
- **Confusion Matrix**: A heatmap showing true vs. predicted labels to evaluate model performance visually.

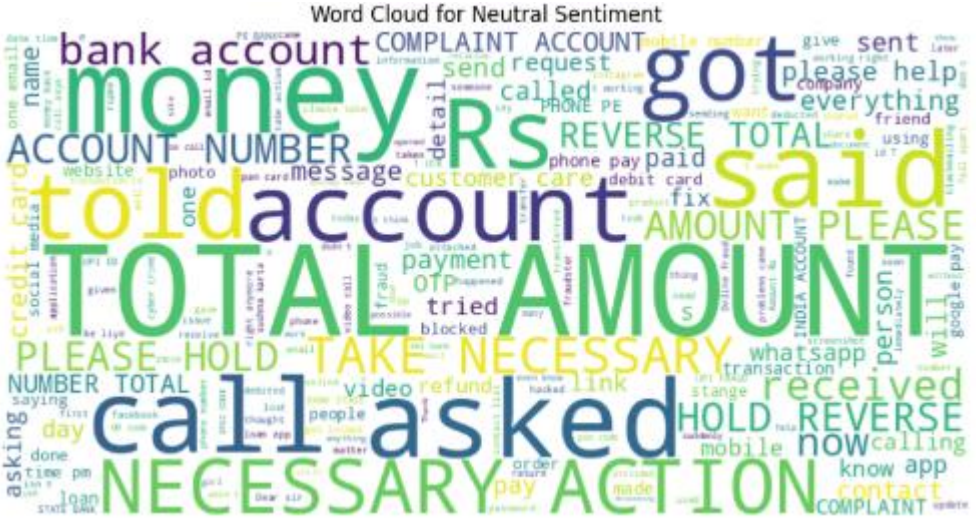


- **Precision, Recall, and F1-Score and Accuracy**: Bar plots highlighting performance for each crime category.

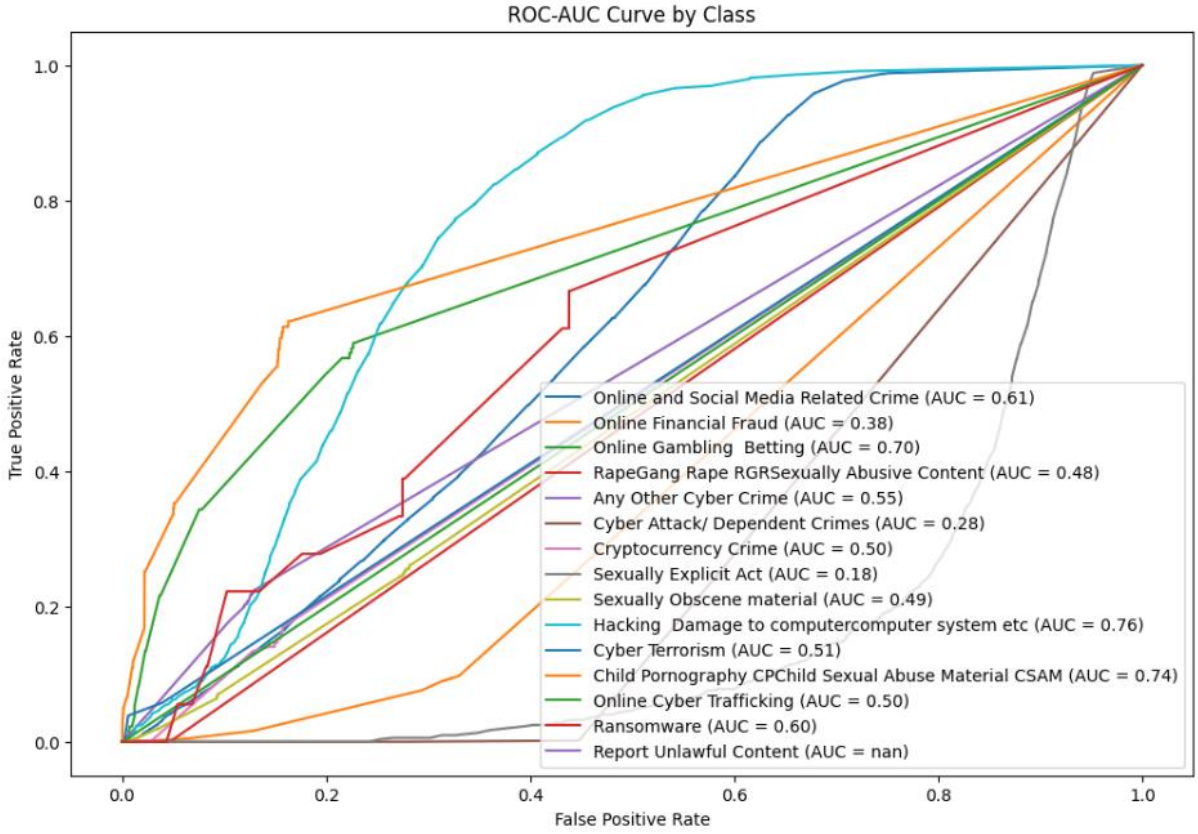


- **Word Cloud**: Visual representation of common words for each sentiment or category.

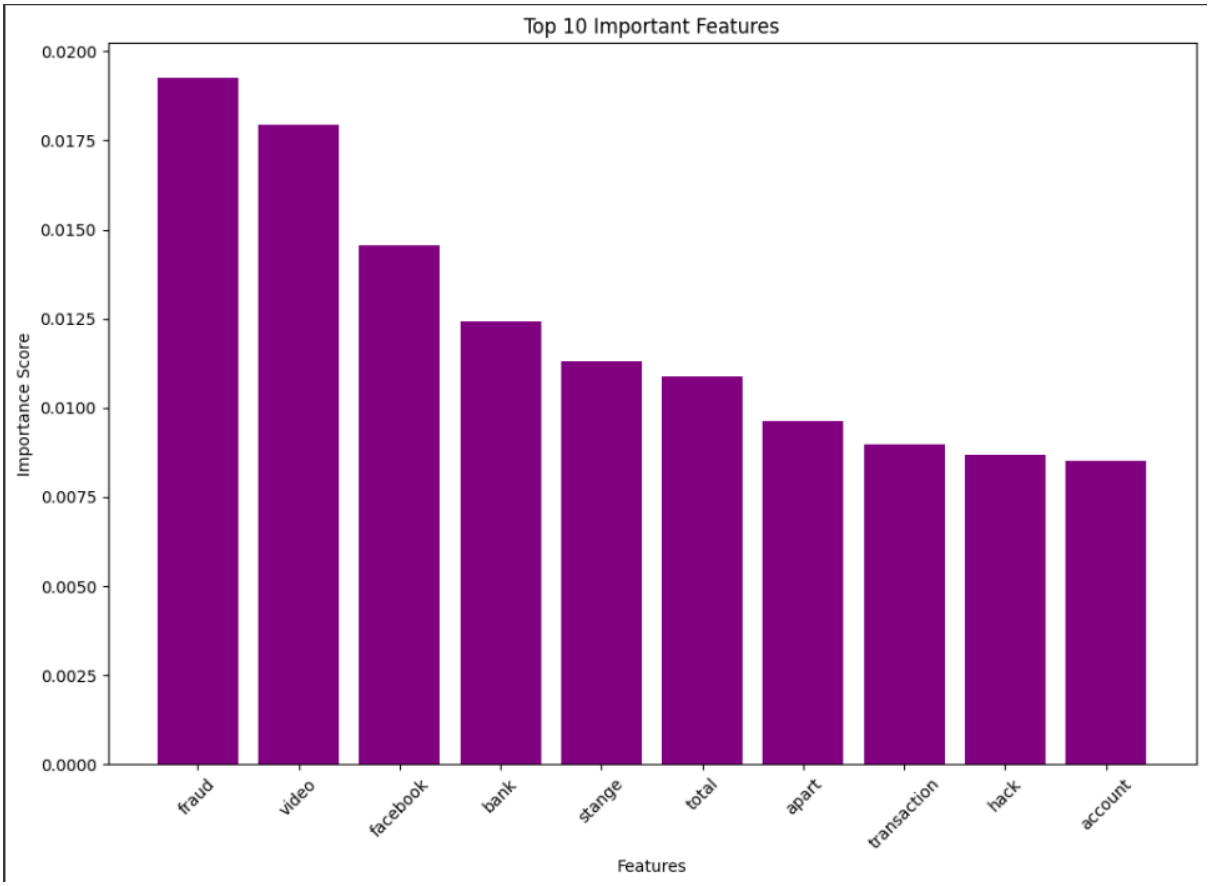




- **ROC-AUC Curve**: Line plots showing the trade-off between true positive and false positive rates for each class.



- **Feature Importance**: Bar graph depicting the top influential features (words) identified by the model.



References and Plagiarism Declaration :

This project was developed using key libraries and tools from the fields of natural language processing (NLP) and machine learning (ML). The following references highlight the frameworks, tools, and methodologies employed to ensure a comprehensive and accurate solution:

References to Tools and Libraries:

1. **Pandas**: Used for data manipulation and pre-processing, facilitating efficient handling of structured datasets.
2. **NumPy**: Provided numerical operations for processing arrays and managing large datasets effectively.
3. **NLTK (Natural Language Toolkit)**: Utilized for text processing tasks such as tokenization, stopword removal, and stemming, essential for preparing textual data.
4. **Scikit-learn**: Enabled machine learning workflows, including label encoding, splitting datasets into training and testing sets, and evaluating model performance through metrics such as accuracy, precision, recall, and F1-score.
5. **Regular Expressions (re module)**: Applied for text cleaning tasks, such as removing punctuation, numbers, and irrelevant symbols, ensuring a clean and consistent input for ML models.
6. **Matplotlib and Seaborn**: Used for generating visualizations to enhance data exploration and evaluation of model performance.
7. **WordCloud**: Assisted in visualizing the most frequent words in the dataset, providing insights into key trends in the data.
8. **SpaCy**: Offered advanced text preprocessing capabilities, including lemmatization and named entity recognition.

Originality and Plagiarism-Free Assurance:

This submission represents the result of my original efforts. All methodologies, implementations, and analyses were designed and executed independently while adhering to best practices in machine learning and NLP. External libraries and tools used in this project are duly acknowledged and cited.

Declaration of Integrity:

I affirm that no plagiarized content has been included in this report. The project adheres to academic and professional integrity standards, ensuring originality in design and execution. Citations for any external work or contributions have been included appropriately to maintain transparency and accountability.

This report, along with its accompanying artifacts, reflects an ethical and professional approach to leveraging modern data science and machine learning techniques.