EXPLORING SENTIMENT AND UNCOVERING OPINIONS IN SOCIAL MEDIA

Minor project-II report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

S SAI	(21UECS0539)	**(VTU19284)**
N PRANAY REDDY	(21UECS0413)	**(VTU19470)**
K M RAMYA SREE	(21UECS0299)	**(VTU19249)**

Under the guidance of
Mr R VINOTH KUMAR,M.Tech.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

May, 2024

EXPLORING SENTIMENT AND UNCOVERING OPINIONS IN SOCIAL MEDIA

Minor project-II report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

S SAI	(21UECS0539)	**(VTU19284)**
N PRANAY REDDY	(21UECS0413)	**(VTU19470)**
K M RAMYA SREE	(21UECS0299)	**(VTU19249)**

Under the guidance of
Mr R VINOTH KUMAR,M.Tech.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

May, 2024

CERTIFICATE

It is certified that the work contained in the project report titled " EXPLORING SENTIMENT AND UNCOVERING OPINIONS IN SOCIAL MEDIA " by " S SAI (21UECS0539), N PRANAY REDDY (21UECS0413), K M RAMYA SREE (21UECS0299) " has been carried out under our supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2024

Signature of Professor In-charge
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2024

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)
S SAI
Date: / /

(Signature)
N PRANAY REDDY
Date: / /

(Signature)
K M RAMYA SREE
Date: / /

APPROVAL SHEET

This project report entitled EXPLORING SENTIMENT AND UNCOVERING OPINIONS IN SOCIAL MEDIA by S SAI (21UECS0539), N PRANAY REDDY (21UECS0413), K M RAMA SREE (21UECS0299) is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Mr R VINOTH KUMAR, M.Tech.,
ASSISTANT PROFESSOR.

Date: / /
Place:

ACKNOWLEDGEMENT

We express our deepest gratitude to our respected **Founder Chancellor and President Col. Prof. Dr. R. RANGARAJAN B.E. (EEE), B.E. (MECH), M.S (AUTO),D.Sc., Foundress President Dr. R. SAGUNTHALA RANGARAJAN M.B.B.S.** Chairperson Managing Trustee and Vice President.

We are very much grateful to our beloved **Vice Chancellor Prof. S. SALIVAHANAN,** for providing us with an environment to complete our project successfully.

We record indebtedness to our **Professor & Dean, Department of Computer Science & Engineering, School of Computing, Dr. V. SRINIVASA RAO, M.Tech., Ph.D.,** for immense care and encouragement towards us throughout the course of this project.

We are thankful to our **Head, Department of Computer Science & Engineering, Dr.M.S. MURALI DHAR, M.E., Ph.D.,** for providing immense support in all our endeavors.

We also take this opportunity to express a deep sense of gratitude to our **Internal Supervisor Mr R VINOTH KUMAR,M.Tech,Assistant Professor.,** for his cordial support, valuable information and guidance, he helped us in completing this project through various stages.

A special thanks to our **Project Coordinators Mr. V. ASHOK KUMAR, M.Tech., Ms. U.HEMAVATHI, M.E., Ms. C. SHYAMALA KUMARI, M.E.,** for their valuable guidance and support throughout the course of the project.

We thank our department faculty, supporting staff and friends for their help and guidance to complete this project.

S SAI	**(21UECS0539)**
N PRANAY REDDY	**(21UECS0413)**
K M RAMYA SREE	**(21UECS0299)**

ABSTRACT

Sentiment analysis on social media is a natural language processing technique used to extract subjective information and opinions from user-generated content on various social media platforms, such as Twitter, Facebook, and Instagram. The goal of this project is to perform sentiment analysis on social media data related to a particular topic or brand, such as a product launch or a social issue. Social media data will be collected using relevant APIs or web scraping tools and pre processed by cleaning and filtering out irrelevant or spam content. A sentiment analysis model, such as a lexicon-based or machine learning model, will be applied to classify the sentiment of the content as positive, negative, or neutral. Results will be visualized and analyzed using various techniques and tools, such as word clouds, bar charts, and time series analysis, to gain insights and make data-driven decisions based on public opinion Challenges in social media sentiment analysis include the use of slang, emojis, and hashtags, as well as the need to handle multilingual content. Sentiment analysis on social media can be a powerful tool for understanding public opinion, customer satisfaction, and brand reputation, and making data-driven decisions in various domains, such as marketing, politics and social sciences.

Keywords:

Natural language processing, Application programming interface, Filtering, Lexicon, Sentiment Analysis, Socialmedia, Webscraping,

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

ACRONYMS	ABBREVIATION
API	Application Programming Interface
CNN	Convolutional Neural Networks
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
RNN	Recurrent Neural Networks
SVM	Support Vector System

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

In the ever-evolving landscape of social media, where billions of users worldwide engage in a continuous exchange of thoughts, opinions, and emotions, the ability to understand and analyze the sentiments expressed within this vast sea of data has become increasingly crucial. This project sets out to explore sentiment analysis, a powerful tool that leverages natural language processing (NLP) techniques to extract and interpret the sentiments embedded in social media content. Platforms such as Facebook, Twitter, and Instagram serve as digital forums where users freely express their views on a wide range of topics, from personal experiences and interests to global events and societal issues. By analyzing this wealth of user-generated content, we can gain valuable insights into the prevailing sentiments, attitudes, and opinions of individuals and communities across the globe. Sentiment analysis involves the use of advanced algorithms and machine learning techniques to classify text into categories such as positive, negative, or neutral, based on the underlying sentiment expressed. Through sentiment analysis, we can uncover the underlying emotions that drive user behavior, providing valuable insights for businesses, researchers alike.

For businesses, sentiment analysis can help gauge customer satisfaction, identify emerging trends, and tailor marketing strategies to better meet consumer needs. For researchers, sentiment analysis offers a window into public opinion and sentiment towards various topics, providing valuable insights for social and behavioral studies. For policymakers, sentiment analysis can inform decision-making processes by providing a deeper understanding of public sentiment towards policies, initiatives, and public figures. However, sentiment analysis in social media is not without its challenges. The informal and often noisy nature of social media text, which is rife with slang, abbreviations, and grammatical errors, can pose challenges to accurate sentiment analysis.. By harnessing the power of sentiment analysis, we can gain a deeper understanding of human behavior, societal trends, and the dynamics of online communities.

1.2 Aim of the project

The project aims to analyze sentiments expressed in social media posts using advanced NLP and machine learning techniques. It seeks to categorize sentiments as positive, negative, or neutral to uncover trends and insights into user opinions. The goal is to provide valuable information for businesses, researchers, and policymakers to improve decision-making and understand public sentiment on various topics.

1.3 Project Domain

The domain of the project revolves around natural language processing (NLP) and machine learning. It encompasses the use of computational techniques to analyze and interpret the sentiments expressed in user-generated content on various social media platforms. This domain involves the development of algorithms and models to categorize sentiments as positive, negative, or neutral, as well as the exploration of techniques to handle challenges such as slang, sarcasm, and context-specific language. The project aims to extract meaningful insights from social media data, providing valuable information for businesses, researchers, and policymakers to understand public opinion, trends, and attitudes towards different topics and events in the digital world.

1.4 Scope of the Project

The scope of the project includes developing and implementing algorithms to analyze sentiments expressed in user-generated content on platforms like Facebook, Twitter, and Instagram. This involves collecting and preprocessing data, applying machine learning and natural language processing techniques for sentiment classification, and exploring the impact of contextual factors on sentiment analysis results. The project aims to provide insights into user opinions, attitudes, and trends, which can be valuable for businesses, researchers, and policymakers. Additionally, the project may involve studying the effectiveness of sentiment analysis in different social media contexts and the development of tools for real-time sentiment monitoring.

Chapter 2

LITERATURE REVIEW

1. In 2020, Kumar et al., [6] have presented a hybrid deep learning approach named ConVNet-SVMBoVW that dealt with the real-time data for predicting the fine-grained sentiment. In order to measure the hybrid polarity, an aggregation model was developed. Moreover, SVM was used for training the BoVW to forecast the sentiment of visual content. Finally, it was concluded that the suggested ConvNet-SVMBoVW was outperformed by the conventional models.
2. In 2020, Xu et al., [13] have introduced a NB method for multi-domain and large-scale E-commerce platform product review classification of sentiment. Consequently, the parameter evaluation method was extended in NB for continuous learning fashion. Later, for fine-tuning the learned distribution on the basis of three types of assumptions, many ways were introduced for acquiring the best performance. The results have shown that the suggested model has high accuracy in Amazon product and movie review sentiment datasets.
3. In 2020, Xu et al., [14] have introduced a NB method for multi-domain and large-scale E-commerce platform product review classification of sentiment. Consequently, the parameter evaluation method was extended in NB for continuous learning fashion. Later, for fine-tuning the learned distribution on the basis of three types of assumptions, many ways were introduced for acquiring the best performance. The results have shown that the suggested model has high accuracy in Amazon product and movie review sentiment datasets.
4. In 2019., Feizollah et al., [4] have concentrated on tweets related to two halal products such as halal cosmetics and halal tourism. By utilizing Twitter search function, Twitter information was extracted, and a new model was employed for data filtering. Later, with the help of deep learning models, a test was performed for computing and evaluating the tweets. Moreover, for enhancing the accuracy and building prediction methods, RNN, CNN, and LSTM were employed. From the outcomes, it was seemed that the combination of LSTM and CNN attained the best accuracy.

5. In 2019.,Yousif et al., [12] have presented a multi-task learning method on the basis of CNN and RNN. The structure of the suggested method was helpful for denoting the citation context and feature extraction was done in an automatic way. By considering two freely accessible datasets, the suggested technique was analyzed. The outcomes have shown that the proposed model was improved than conventional models.

6. In 2019.,Vashishtha and Susan [11] have calculated the sentiment related to social media posts by a new set of fuzzy rules consisting of many datasets and lexicons. The developed model combined Word Sense Disambiguation and NLP models with a new unsupervised fuzzy rule-based model for categorizing the comments into negative, neutral, and positive sentiment class. The experiments were performed on 3 sentiment lexicons, four existing models, and nine freely available twitter datasets. The outcomes have shown that the introduced method was attaining the best results.

7. In 2019, Park et al., [10] have developed a semi-supervised sentiment-discriminative objective for resolving the issue by documents partial sentiment data. The suggested model not only reflected the partial data, but also secured the local structures obtained from real data. The suggested model was evaluated on real time datasets. The results have shown that the suggested model was performing well.

8. In 2019., Ray and Chakrabarti [8] have introduced a deep learning algorithm for extracting the features from text and the user's sentiment analysis with respect to the feature. In opinionated sentences, a seven layer Deep CNN was employed for tagging the features. In order to enhance the performance of sentiment scoring and feature extraction models, the authors merged the deep learning methods using a set of rule-based models. Finally, it was seen that the suggested method achieved the best accuracy.

9. In 2019, Zhao et al., [9] have offered a novel image-text consistency driven multi-modal sentiment evaluation model, which explored the correlation among the text and image. Later, a multi-modal adaptive sentiment analysis model was implemented. By using the traditional SentiBank model, the mid-level visual features were extracted and those were employed for representing the International Journal of Advanced Science and Technology Vol. 29, No. 7, (2020), pp. 1462-1471 1465 ISSN: 2005-4238 IJAST Copyright 2020 SERSC visual the-

ories by integrating the different characteristics like social, textual, and visual features for introducing a machine learning model. The suggested model has attained best performance when compared over traditional models.

10. In 2019, Afzaal et al., [3] have recommended a novel approach of aspect-based sentiment classification, which recognized the features in a precise manner and attained the best classification accuracy. Moreover, the scheme was developed as a mobile application, which assisted the tourists in identifying the best hotel in the town, and the proposed model was analyzed using the real-world data sets. The results have shown that the presented model was effective in both recognition as well as classification.

11. In 2019., Saad and Yang [1] have aimed for giving a complete tweet sentiment analysis on the basis of ordinal regression with machine learning algorithms. The suggested model included pre-processing tweets as first step and with the feature extraction model, an effective feature was generated. The methods such as SVR, RF, Multinomial logistic regression (SoftMax), and DTs were employed for classifying the sentiment analysis. Moreover, twitter dataset was used for experimenting the suggested model. The test results have shown that the suggested model has attained the best accuracy, and also DTs were performed well when compared over other methods.

12. In 2018, Abdi et al., [7] have proffered a machine learning technique for summarizing the opinions of the users mentioned in reviews. The suggested method merged multiple kinds of features into a unique feature set for modelling accurate classification model. Therefore, a performance investigation was done for four best feature selection models for attaining the best performance and seven classifiers for choosing the relevant feature set and recognized an effective machine learning algorithm. The suggested method was implemented in various datasets. The outcomes have demonstrated that the combination of IG as the feature selection approach and SVM-based classification approach enhanced the performance.

13. In 2018, Mukhtar et al., [5] have performed the sentiment analysis to the Urdu blogs attained from several domain with Supervised Machine learning and Lexicon-based models. In Lexicon-based models, a well-performing Urdu sentiment analyzer and an Urdu Sentiment Lexicons were employed, whereas, in Supervised Machine learning algorithm, DT, KNN, and SVM were employed.

The data were combined from the two soruces for performing the best sentiment analysis. Based on the tests conducted, the outcomes were shown that the Lexicon-based model was superior to the supervised machine learning algorithm.

14. Fang et al., [2] have suggested multi-strategy sentiment analysis models using the semantic fuzziness for resolving the issues. The outcomes have demonstrated that the proposed model has attained high efficiency.

15. In 2018, Smadi et al., [15] have proposed existing models on the basis of supervised machine learning algorithms for specifying the defects of feature-based sentiment analysis of Arabic hotel's review. Moreover, SVM and Deep RNN were developed and trained with word, lexical, morphological, semantic, and syntactic features. The reference dataset of Arabic hotel's review dataset was used for evaluating the proposed model. The outcomes have shown that SVM was performing well when compared over RNN model.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

The people in the society show a lot of interest to find other's opinion about themselves or any product or any topic. So in the present days people keep on posting any topic related to them in social media platforms like Instagram, Facebook and so on to know other's opinion about that topic. So the people view that posted topic and may comment their opinion about the posted topic. The person who posted the topic need to view all the comments posted by the users to take a decision or to know about the review of that posted topic. This process may take more time.

Disadvantages

- Cannot know the correct review about topic/product.
- Takes longer time to read all the reviews.
- Owner cannot get accurate idea about user's reviews.

3.2 Proposed System

The technique involved in this process is Sentiment Analysis.The system will utilize APIs to collect user-generated content from various social media platforms, clean and preprocess the data to remove noise and standardize the text format, and then apply advanced NLP and machine learning techniques to classify sentiments as positive, negative, or neutral. The sentiment analysis results will be visualized using interactive charts or graphs, providing users with actionable insights into the sentiments expressed on social media. The system will be designed for scalability and performance, ensuring that it can handle large volumes of data efficiently and provide real-time analysis capabilities.

Advantages

- Designed for scalability to handle large data volumes efficiently.
- Offers real-time analysis capabilities.

- Time efficient.
- More accurate.
- Can avoid reading all the comments posted by users.

3.3 Feasibility Study

Developing a simple sentiment analysis model may indeed take some time, but it can be a valuable and useful tool in various applications. Sentiment analysis has wide-ranging applications in understanding customer feedback, social media monitoring, and even analyzing public opinion on various topics. By developing a simple model, you can gain valuable insights into the sentiment expressed in text data, which can help in making informed decisions and improving user experiences. Additionally, starting with a simple model allows for easier implementation and understanding of the underlying concepts, making it a good starting point for further exploration and development in the field of natural language processing.

3.3.1 Economic Feasibility

The model does not require expensive or rare resources, making it economically feasible. Since it is a software model, there is no need to spend money on specialized hardware or tools. This makes it accessible to a wide range of users, including those with limited budgets. Additionally, the simplicity and ease of the project contribute to its economic feasibility. The project does not involve complex algorithms or extensive data processing, reducing the time and resources required for development and implementation. This also means that the project can be completed quickly and efficiently, further lowering costs. It demonstrates that valuable insights can be gained from sentiment analysis without the need for expensive investments, making it an attractive option for businesses and individuals alike.

3.3.2 Technical Feasibility

This model is technically feasible, requiring only a basic knowledge of the technical field. While some familiarity with machine learning algorithms and Python programming is necessary, deep technical expertise is not required. The project is primarily based on simple machine learning algorithms, which are widely used and well-documented. This makes it easier for individuals with basic knowledge in this domain to develop the project. Additionally, Python is a popular programming language for machine learning, known for its readability and ease of use, further contributing to the project's technical feasibility. The technical feasibility of the sentiment analysis project lies in its reliance on widely used and accessible technologies. With the right foundational knowledge and resources, individuals with basic technical skills can successfully develop and implement the project..

3.3.3 Social Feasibility

The model is socially feasible, as it addresses a growing need for such models in today's society. With the increasing use of social media platforms, people are more active than ever in sharing their opinions and sentiments online. As a result, there is a growing demand for tools that can help analyze and understand these sentiments.By providing a way to automatically categorize comments based on their sentiment, the project meets a practical need for individuals and businesses looking to gain insights from social media data. This makes it socially relevant and valuable in today's digital age. Furthermore, the project's applicability to social media platforms makes it accessible to a wide audience, further enhancing its social feasibility. Overall, the sentiment analysis project is socially feasible due to its relevance and applicability in today's social media-driven society.

3.4 System Specification

3.4.1 Hardware Specification

- **Hard Disk** : 5 GB
- **Processor** : i5
- **ROM** : 1TB
- **Memory (RAM)**: At least 8 GB of RAM to ensure smooth performance when processing and analyzing data
- **Operating System**: A 64-bit operating system

3.4.2 Software Specification

- **Preprocessing**: NLTK

- **Visualization**: Matplotlib and Seaborn for data visualization.
- **Data Collection**: Tweepy for Twitter data collection.

3.4.3 Standards and Policies

Colab:

Google Colab, short for Google Colaboratory, is a cloud-based platform provided by Google that offers free access to nlp resources for running and executing Python code. It is primarily designed for machine learning and data analysis tasks, providing an environment where users can write and execute Python code through a web-based interface without requiring any local setup.

- **Standard Used:ISO/IEC 27001**

ISO/IEC 27001 plays a crucial role in ensuring the security and integrity of the data involved. The standard provides guidelines for managing information security risks, which is essential when dealing with sensitive data from social media platforms. ISO/IEC 27001 emphasizes the need for organizations to protect data from unauthorized access, maintain data privacy, and ensure the reliability of the analysis results.

Jupyter

It's like an open source web application that allows us to share and create the documents which contains the live code, equations, visualizations and narrative text. It can be used for data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning.

Chapter 4

METHODOLOGY

4.1 Architecture Diagram

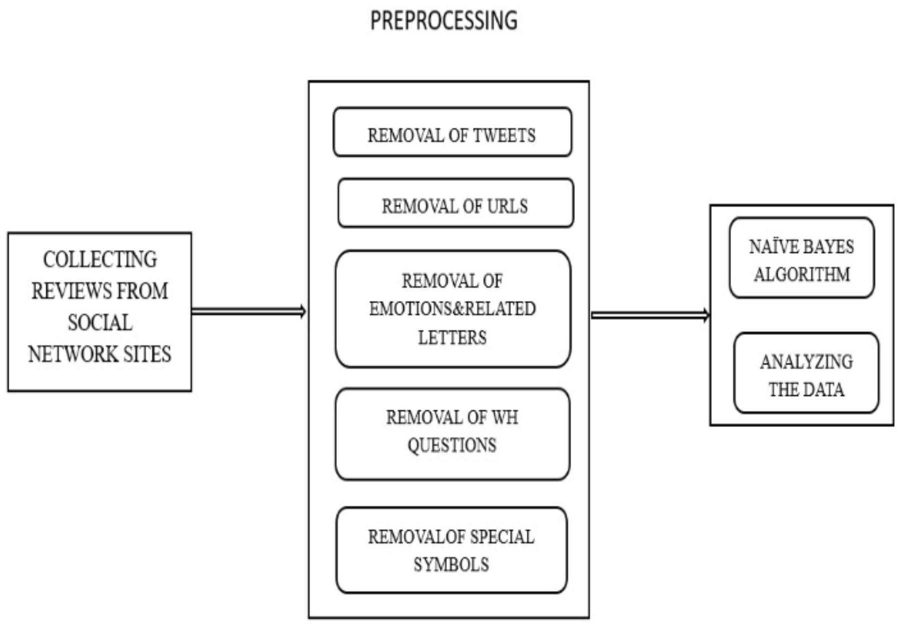


Figure 4.1: **Architecture diagram for sentiment analysis**

The model initially collect the data/reviews like tweets, reviews, comments, emoji's from social networking sites.The data is pre-processed by removing retweets. Then removing URLs. Then removing the 'WH' questions.Then last step in the data pre-processing removing of special symbols. Now, supervised learning algorithms are applied on the filtered data. The Navie Bayes algorithm Analyze the data and predicts the probability of kind of tweets, comments, reviews, comments present in the data and gives the output.

4.2 Design Phase

4.2.1 Data Flow Diagram

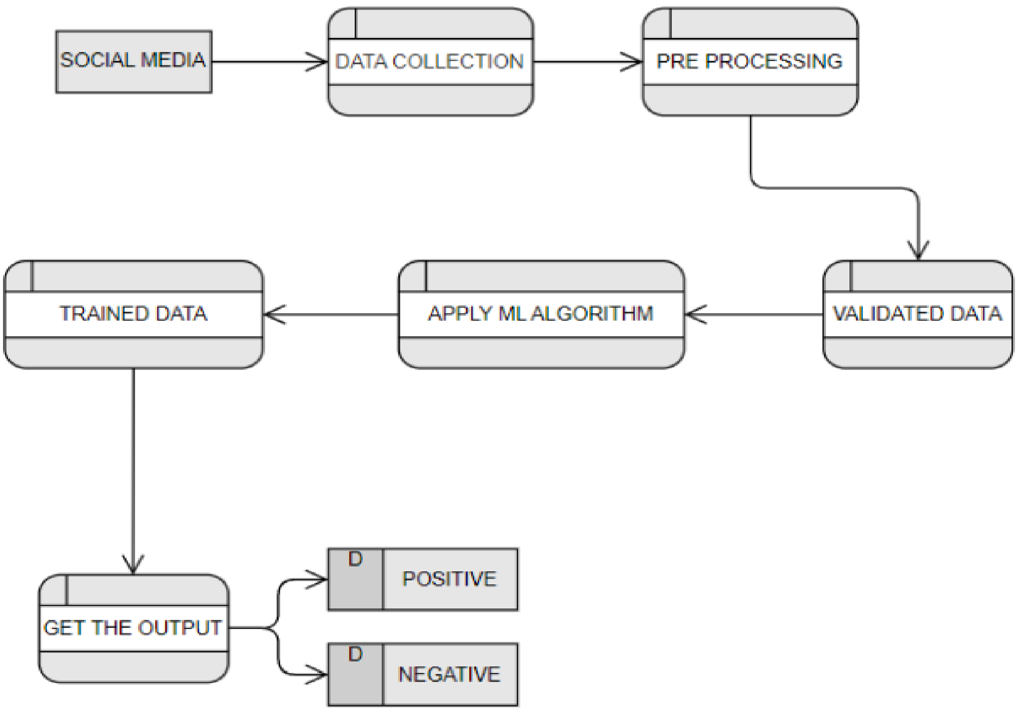


Figure 4.2: **Data flow diagram for sentiment analysis**

The system first collects the data. Then the sentiment is extracted from the tweets and it will be given to the training set. The training set actually contains the sentiment words with positivity and negativity. So,the training set determines the sentiment in the tweet, pre-processed, and grade them. Algorithm that we used predicts the probability and classify the tweet as a negative one or positive one.

4.2.2 Use Case Diagram

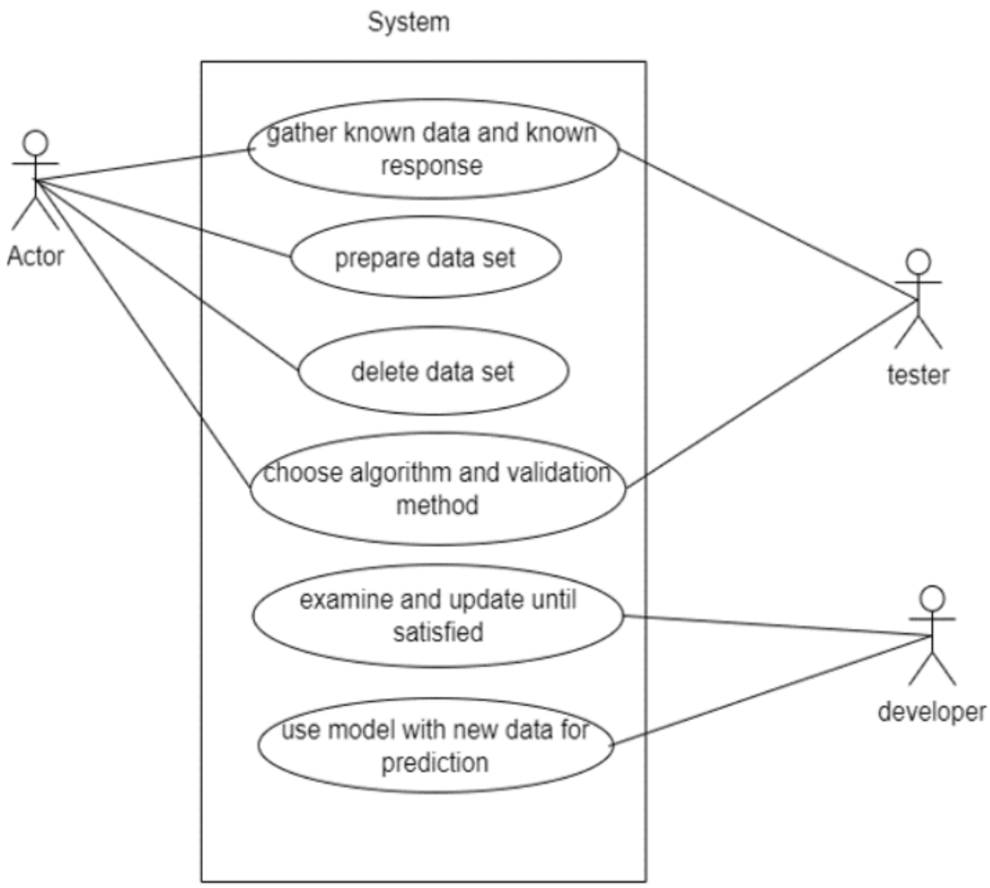


Figure 4.3: **Usecase Diagram for sentiment analysis**

The user in this system posts a topic/item to collect its review. Then the users post their review in the comments. The system then extracts the sentiment in the comments and determines the positivity and negativity in the tweets. Based on that it classifies the tweets and post a final review about the topic. So, it's a method to collect the review about the topic based on the comments of users through opinion mining.

4.2.3 Class Diagram

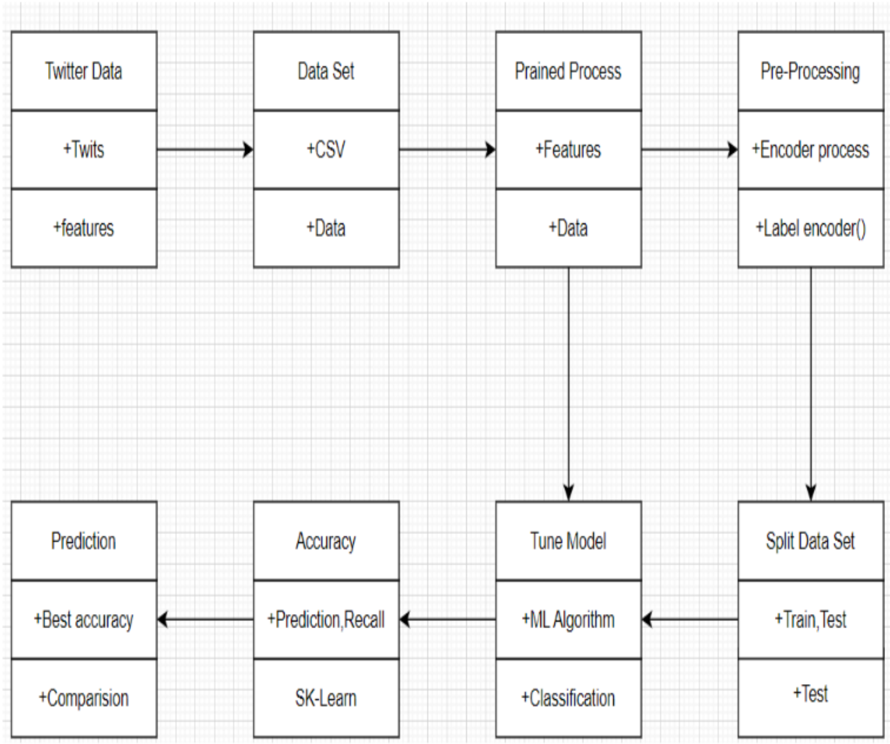


Figure 4.4: **Class diagram for sentiment analysis**

The above picture depicts the different classes in opinion mining process following contains the trained dataset, features. Preprocessing: This class contains methods for encoder process and label encoder. Tune Model:This class contains the machine learning algorithm used for classifying the tweets, reviews, emojis. Prediction:This class contains classified data and predicted data.

4.2.4 Sequence Diagram

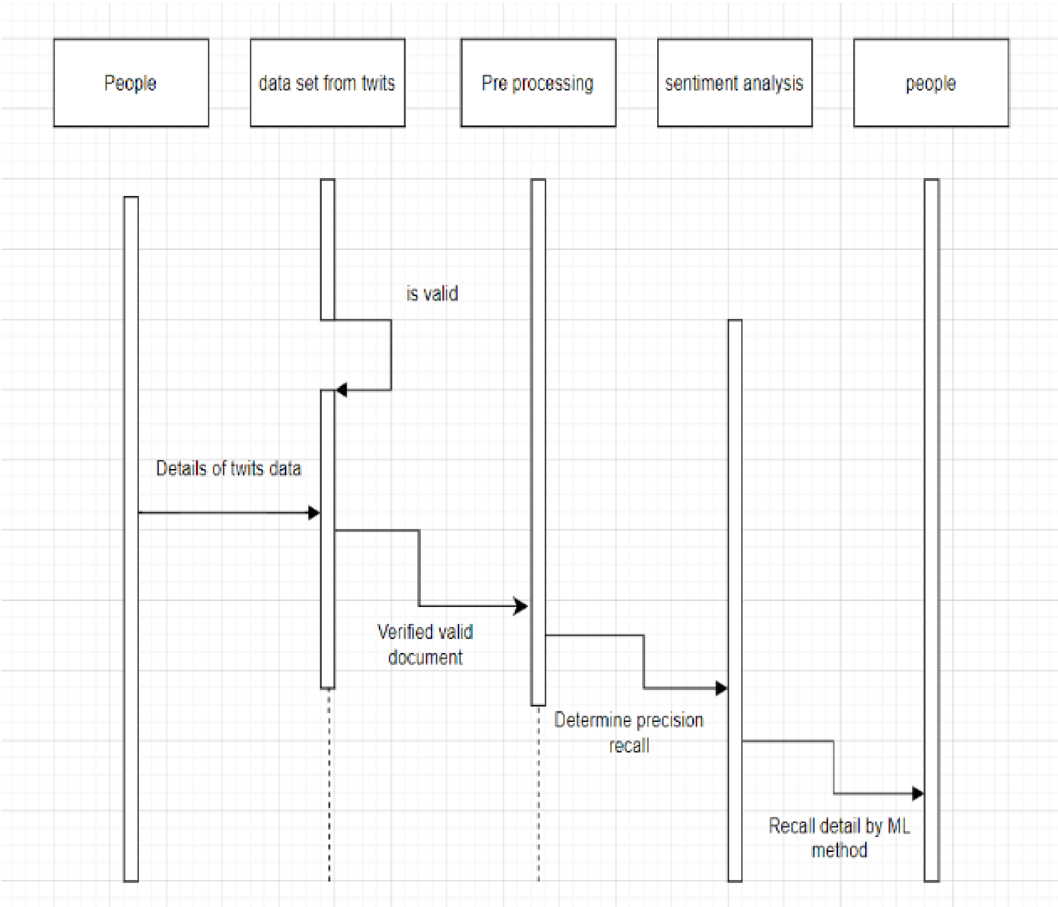


Figure 4.5: **Sequence diagram for sentiment analysis**

The user first logs in to the social network. Then he sees the posted topic and posts his opinion in form of comments. So, the system then extracts all the sentiment keywords (features) in the tweets and checks the positive and negative sentiment based on it's data set. So, then the classification will be done and it provides a review about the posted topic.

4.2.5 Activity Diagram

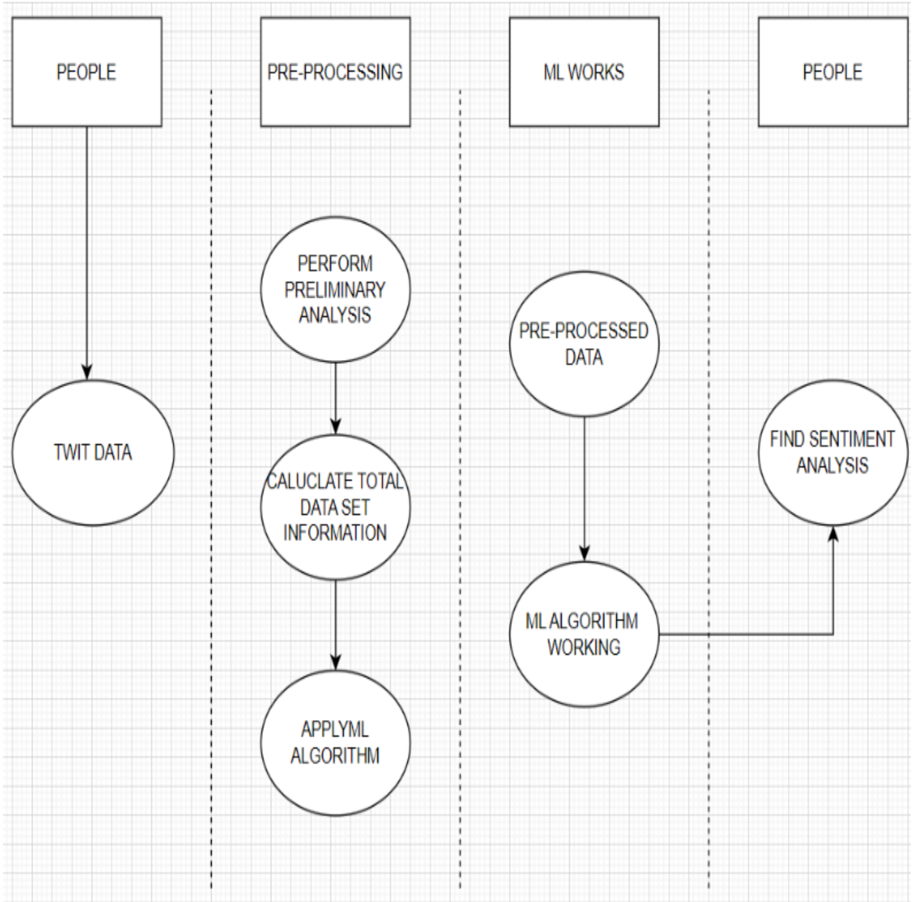


Figure 4.6: **Activity Diagram for sentiment analyis**

The above picture depicts the activity that can be done at each phase. The first phase is the collecting the twits data from the people and transferred to next phase called Pre-Processing in this we perform primary analysis of data and later calculate the total data set information and later we apply Machine learning algorithm concept and its processsed through next phase called Machine learning works in this phase the pre-processed data was worked under the machine learning algorithm and finally the sentimental analysis has been done for data.

4.3 Algorithm & Pseudo Code

4.3.1 Natural Language Processing Algorithm

1. Import all required library files.

- Import pandas for data handling.
- Import nltk for natural language processing tasks.

2. Get the data file to read the reviews.

- Load the dataset containing reviews and their sentiment labels

3. Grade the reviews based on repeated words.
4. Separate the positive and negative,irrelevant words.

- Preprocess the text by removing stopwords, punctuation, and converting words to lowercase.
- Tokenize the text into words.
- Lemmatize words to their base form

5. Train the data.
6. Read the given input and separate into words.
7. Calculate the count of positive and negative words and irrelevant words.

- Use the trained model to predict the sentiment of each word in the input text.
- Count the number of positive, negative, and irrelevant words based on the model's predictions.

4.3.2 Pseudo Code

1. Collect social media posts (tweets, Facebook posts, etc.)
2. Preprocess the text:

a. Remove special characters, hashtags, mentions, URLs
b. Tokenization
c. Lowercasing
d. Remove stop words

3. Use a sentiment lexicon or machine learning model to classify the sentiment of each post:

a. Lexicon-based approach:

i. Assign a polarity score to each word in the post (positive, negative, neutral)

ii. Calculate the overall sentiment score for the post based on the polarity scores of its words

b. Machine learning approach:

i. Train a classifier using labeled data (posts with sentiment labels)

ii. Use the trained classifier to predict the sentiment of new posts

4. Aggregate sentiment scores:

a. Calculate the average sentiment score for all posts

b. Visualize sentiment distribution

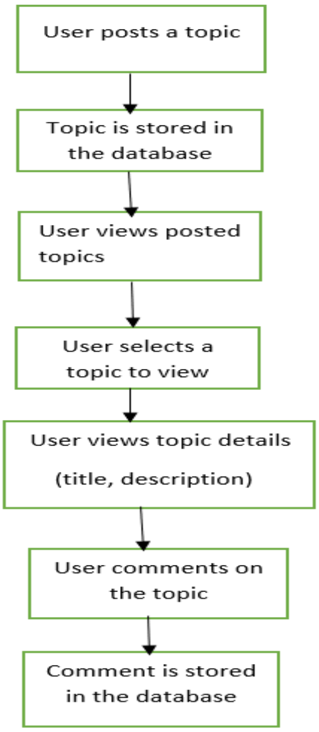
5. Interpret the results and draw conclusions about the overall sentiment on the social media platform.

4.4 Module Description

4.4.1 Data Collection Module

This module is essential for acquiring the text data necessary for analysis. This module typically involves several key steps. Firstly, the project must identify and select the sources from which data will be collected, such as social media platforms (e.g., Twitter, Facebook, Reddit), review websites or other relevant text-based sources. Once the sources are determined, the project can choose the appropriate method for data collection, which may include using APIs provided by the platforms, web scraping techniques to extract data from websites, or utilizing pre-existing datasets.

After selecting the data collection method, the project can proceed with implementing the data collection process. For API-based data collection, this would involve registering for access to the APIs and following their guidelines for data retrieval. For web scraping, the project would need to develop scripts to extract the relevant text data from web pages, ensuring compliance with the terms of service of the websites being scraped. Additionally, the collected data may require preprocessing to clean it and remove any irrelevant information, such as HTML tags, special characters, or duplicate entries. For projects requiring real-time analysis, mechanisms can be implemented to periodically update the dataset with new data from the sources. Throughout the data collection process, it is important to consider legal and ethical considerations, such as respecting the terms of service and privacy policies of the platforms being used, and ensuring that the data is collected and used responsibly.

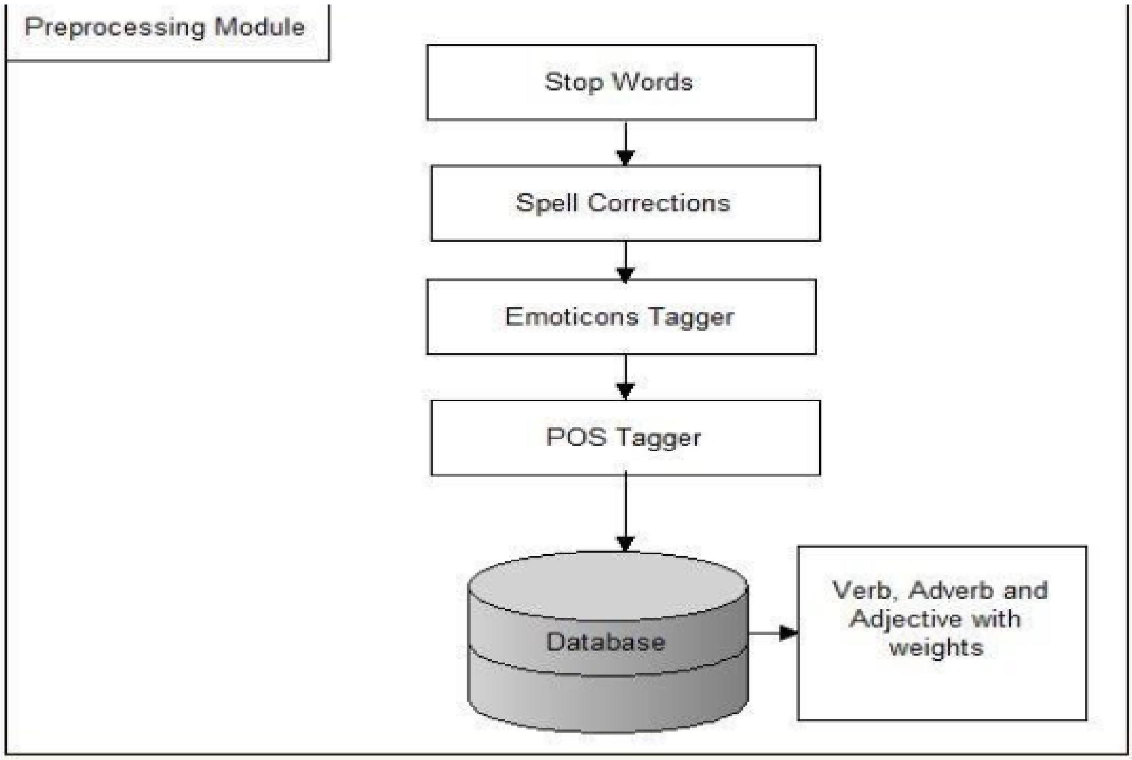


4.4.2 Preprocessing Module

This module is crucial for preparing text data before it can be fed into the analysis model. This module typically includes several key steps to clean and normalize the text data.Firstly, the module may involve converting the text to lowercase to ensure consistency in the analysis, as uppercase and lowercase versions of the same word should be treated as the same word. Next, the module might remove any special characters, such as punctuation marks, numbers, or other non-alphabetic characters, as these are typically not relevant to the sentiment analysis task.

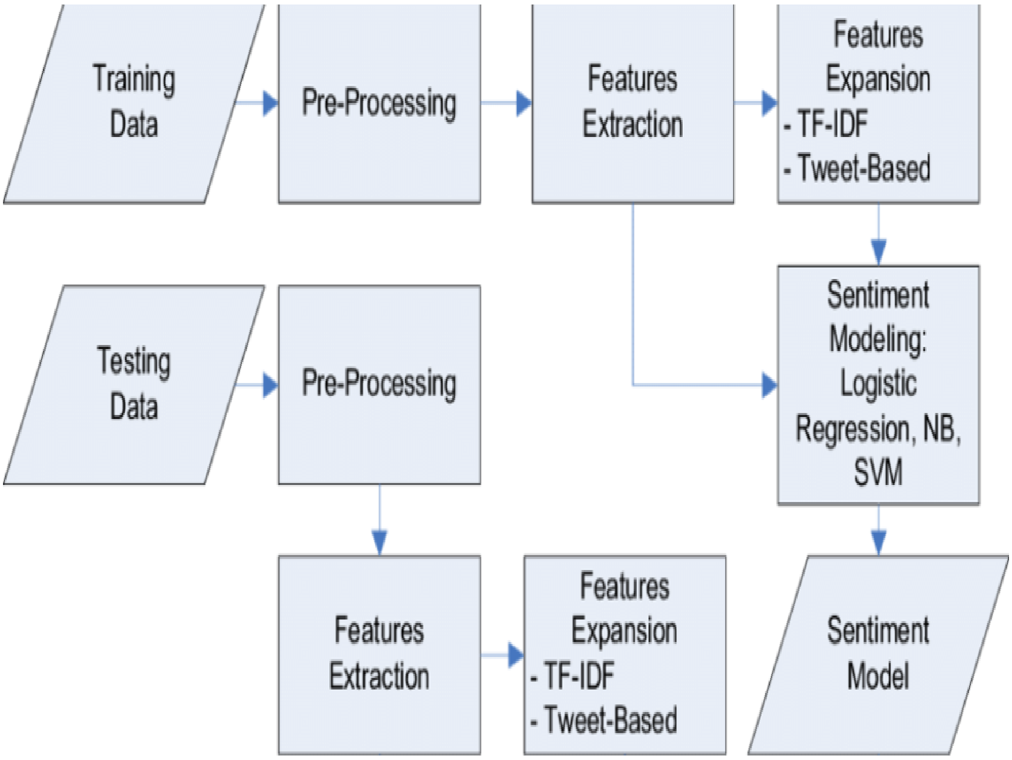
Additionally, the module may handle tokenization, which involves splitting the text into individual words or tokens. This step is important for further processing as it allows the analysis model to work with individual words rather than entire sentences. Stopword removal is another common step in the preprocessing module, where common words that do not carry much meaning, such as ”the,” ”is,” and ”and,” are removed from the text. The module may also include stemming or lemmatization, which involves reducing words to their base or root form. This step helps in reducing the dimensionality of the data and ensuring that different variations of the same word are treated as the same word, which can improve the accuracy of the analysis model.

Finally, the preprocessed text data is ready to be used for sentiment analysis. The preprocessing module plays a crucial role in ensuring that the text data is clean, standardized, and suitable for analysis, ultimately improving the accuracy and effectiveness of the sentiment analysis model.



4.4.3 Feature Extraction Module

The feature extraction module in a sentiment analysis project converts cleaned text data into features that a machine learning model can understand. It uses techniques like Bag-of-Words (BoW) to count the frequency of words in a text, giving more weight to words that appear more often in the document but less in the whole dataset, like ”happy” or ”sad.” Another method is TF-IDF, which calculates the importance of a word in a document relative to how often it appears in other documents, helping to find words specific to a document’s sentiment. Word embeddings like Word2Vec convert words into numerical vectors based on their context in a large dataset, capturing relationships between words like ”good” and ”great.” N-grams group words together, capturing phrases or expressions that might indicate sentiment, such as ”not good.” These techniques transform text into numerical features that can be used to train a sentiment analysis model, helping it understand and predict sentiment in new text data. The choice of technique depends on the complexity of the sentiment analysis task and the nature of the text data being analyzed.

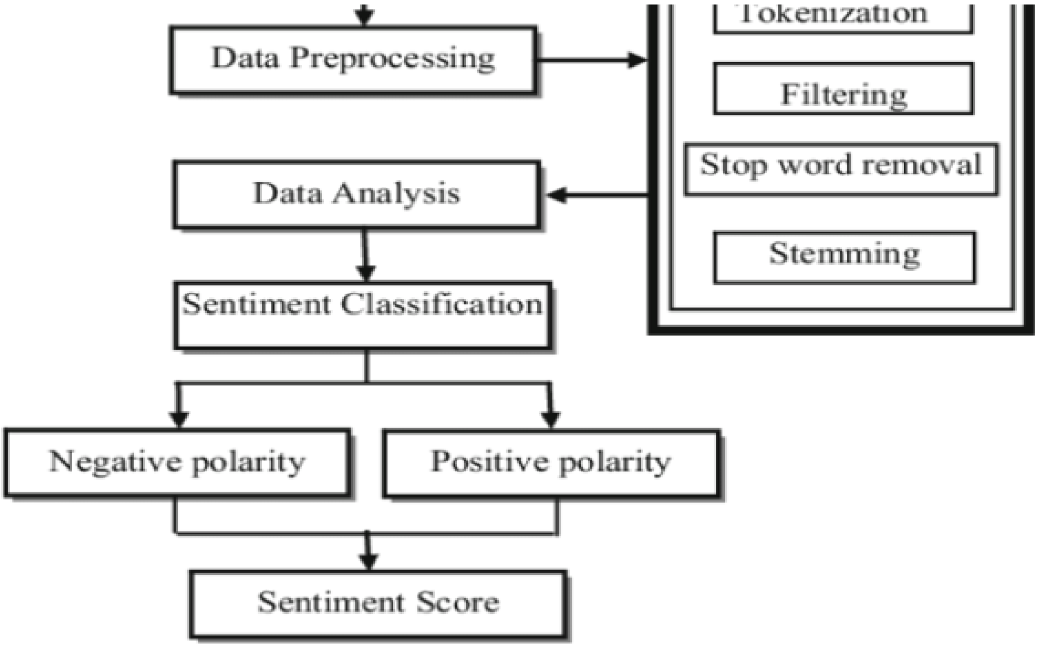


4.4.4 Prediction and Classification Module

The prediction and classification module in a sentiment analysis project is responsible for training a machine learning model to predict the sentiment of text data. This module typically involves several key steps, including data splitting, model selection, training, and evaluation. Firstly, the dataset is split into training, validation, and test sets. The training set is used to train the model, the validation set is used to tune hyperparameters and prevent overfitting, and the test set is used to evaluate the final performance of the model.Next, a suitable machine learning model is selected. Common choices for sentiment analysis include logistic regression, support vector machines (SVM), random forests, and neural networks. The choice of model depends on the complexity of the sentiment analysis task and the nature of the text data.

The selected model is then trained on the training set using the features extracted from the text data. During training, the model learns the patterns and relationships in the data that are indicative of sentiment. After training, the model is evaluated on the validation set to ensure that it generalizes well to new, unseen data. This step helps to fine-tune the model's hyperparameters and prevent overfitting.

Finally, the performance of the trained model is evaluated on the test set to assess its accuracy, precision, recall, and F1-score. These metrics help to determine how well the model performs at predicting the sentiment of text data. Overall, the prediction and classification module plays a crucial role in training and evaluating machine learning models for sentiment analysis, helping to identify the most effective model for the task at hand.



;

4.5 Steps to execute/run/implement the project

4.5.1 Objective Definition

Define the goals and objectives of the sentiment analysis project for social media.

4.5.2 Platform Selection:

Choose the social media platform(s) from which to collect data (e.g., Twitter, Facebook).

4.5.3 Data Collection:

Collect social media posts, comments, or tweets using the platform's API.

4.5.4 Data Preprocessing:

Clean and preprocess the collected data by removing URLs, hashtags, and emojis, and tokenizing the text.

4.5.5 Sentiment Analysis Techniques:

Select the appropriate sentiment analysis techniques, such as lexicon-based or machine learning approaches.

4.5.6 Feature Extraction

Extract features from the preprocessed text data using techniques like Bag-of-Words (BoW) or TF-IDF.

4.5.7 Model Training:

Train a sentiment analysis model on a labeled dataset with positive, negative, and neutral sentiments.

4.5.8 Model Evaluation:

Evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score.

4.5.9 Dashboard Creation:

Create a dashboard to visualize sentiment trends, word clouds, and sentiment distribution.

4.5.10 Deployment and Monitoring:

Deploy the sentiment analysis system to analyze real-time social media data and monitor its performance.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input and Output

5.1.1 Input Design

The dataset of various user reviews from various social media platform had collected and displaying over here.

```
In [1]: import numpy as np # linear algebra
        import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)


        import os
        for dirname, _, filenames in os.walk('/kaggle/input'):
            for filename in filenames:
                print(os.path.join(dirname, filename))

In [2]: import pandas as pd

In [2]:
        df1=pd.read_csv("C:/Users/mm/Downloads/twitter_validation.csv")

In [3]: df2=pd.read_csv("C:/Users/mm/Downloads/twitter_training.csv")

In [4]: df1.head()
```

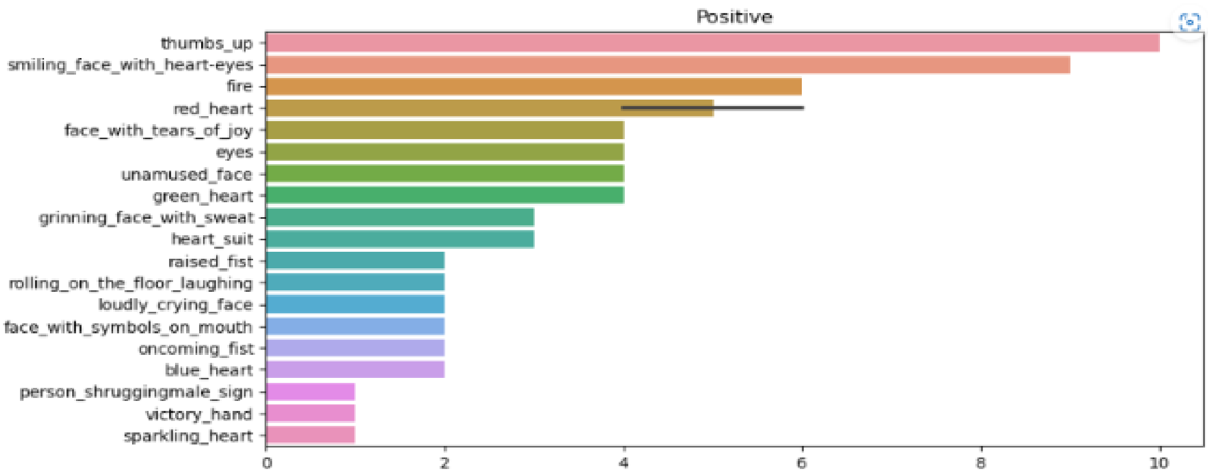
Figure 5.1: **Uploading a Dataset**

5.1.2 Output Design

```
In [35]: target_emojis = {'Positive': [],
                          'Neutral': [],
                          'Irrelevant': [],
                          'Negative': []}

         pattern = re.compile('\u200d')
         for i, text in enumerate(texts):
             emoji_count = split_count(text)
             if emoji_count:
                 emoji_count = [re.sub(pattern, '', e) for e in emoji_count]
                 target_emojis[df1[TARGET].iloc[i]].extend(emoji_count)
```

```
In [36]: for t, emojis in target_emojis.items():
             plt.figure(figsize=(10, 5))
             bar_info = pd.Series(emojis).value_counts()[:20]
             print('=========='*10,  f'\nTop emojis for {t} \n', list(bar_info.index))
             bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
             sns.barplot(x=bar_info.values, y=bar_info.index)

             plt.title(f'{t}')
             plt.show()
```

```
====================================================================================================
Top emojis for Positive
 ['👍', '😍', '🔥', '❤️', '😂', '❤', '👀', '😒', '💚', '😅', '♥', '✊', '🤣', '😭', '🤬', '👊', '💙', '🤷♂', '✌', '💖']
```




```
====================================================================================================
Top emojis for Neutral
 ['😂', '🤔', '😻', '👉', '👈', '❤️', '😩', '🎯', '🤬', '💪', '🎉', '🎥', '😍', '✨', '😐', '🙁', '🔥', '🤘', '😐', '👍']
```

Figure 5.2: **Visual output**

5.2 Testing

Testing is the phase of the project that is done after implementation phase. This testing process is done to check whether the output is coming in our desired manner or not. After testing,if the output is in our required manner,we can conclude that our project is a successful one.

5.3 Types of Testing

5.3.1 Unit testing

Unit testing It is a kind of testing in which, individual or separate components of a software or project are tested. This is to check whether each part of the code performs as expected. This testing is performed at the development phase of the application by developers.

Input

```
df1.columns = ['#', 'refers to', 'sentiment', 'text']
df2.columns = df1.columns
df1['#'].value_counts()
df1['refers to'].value_counts()
```

Test result

```
df_output[["sentiment_score"]].describe()
```

```
In [14]: df_output[["sentiment_score"]].describe()
Out[14]:
```

	sentiment_score
count	300.000000
mean	0.594633
std	0.376298
min	-0.895700
25%	0.440400
50%	0.715900
75%	0.865800
max	0.988300

```
In [ ]:
```

Figure 5.3: **Sentiment Score**

5.3.2 Integration testing

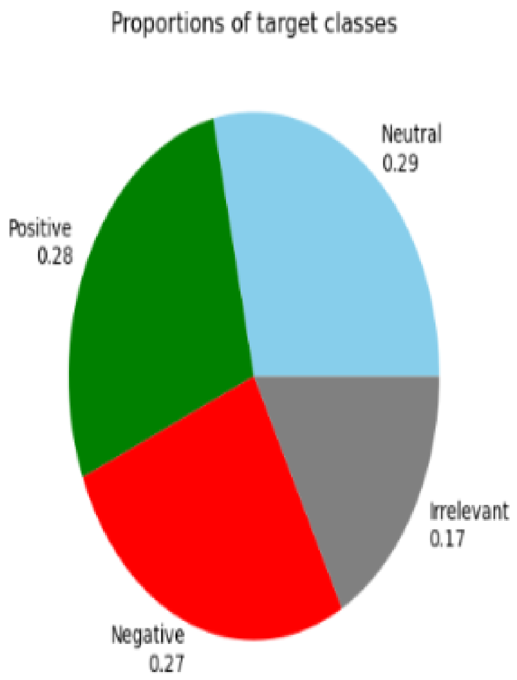
It is a kind of testing in which, we combine the individual parts of the code and test as a group. This testing's purpose is to provide faults in the interaction between the combined parts. Test drives and Test stubs are used in the integration testing process.

Input

```
df1.columns = ['#', 'refers to', 'sentiment', 'text']
df2.columns = df1.columns
df1['#'].value_counts()
df1['refers to'].value_counts()
```

Test result

```
In [25]: target_balance = df1[TARGET].value_counts()

         plt.figure(figsize=(5, 5))
         plt.pie(target_balance, labels=[f'{i}\n{round(target_balance[i]/len(df1), 2)}' for i in target_balance.index],
                 colors=['skyblue', 'g', 'red', 'grey'])
         plt.title('Proportions of target classes')
         plt.show()
```



5.3.3 System testing

It is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements. In system testing, integration testing passed components are taken as input.

Input

```
import re
from nltk.corpus import stopwords
stopwords_list = stopwords.words('english')

word_counts = {'Positive': [],
               'Neutral': [],
               'Irrelevant': [],
               'Negative': []}
  for t, emojis in target_emojis.items():
    plt.figure(figsize=(20, 15))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10, f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.boxplot(x=bar_info.values, y=bar_info.index)
    plt.title(f'{t}')
    plt.show()
```

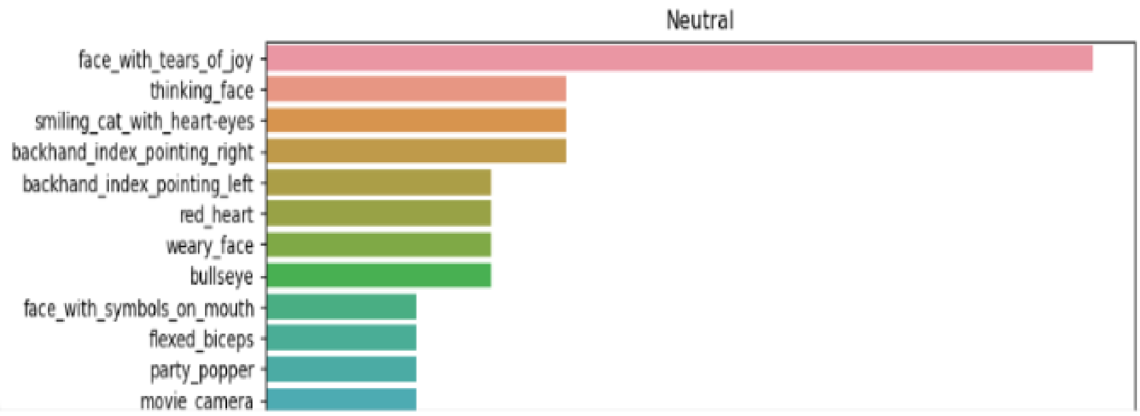
5.3.4 Test Result

The emojis that are identified and classified to positive,negative,neutral,irrelevant and represented them in bar graph.

```
target_emojis = {'Positive': [],
                 'Neutral': [],
                 'Irrelevant': [],
                 'Negative': []}

pattern = re.compile('\u200d')
for i, text in enumerate(texts):
    emoji_count = split_count(text)
    if emoji_count:
        emoji_count = [re.sub(pattern, '', e) for e in emoji_count]
        target_emojis[df1[TARGET].iloc[i]].extend(emoji_count)
```

```
for t, emojis in target_emojis.items():
    plt.figure(figsize=(10, 5))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10,  f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.barplot(x=bar_info.values, y=bar_info.index)

    plt.title(f'{t}')
    plt.show()
```

```
====================================================================================================
Top emojis for Neutral
 ['😂', '🤔', '😻', '👉', '👈', '❤', '😩', '🎯', '🤬', '💪', '🎉', '🎥', '😍', '✨', '😑', '😳', '🔥', '🤘', '0',
'👍']
```




```
In [37]: plt.figure(figsize=(8, 10))
         sns.heatmap(pd.crosstab(df1['refers to'], df1[TARGET], normalize='index'), annot=True)
         plt.title('Frequencies of meeting referred objects in each category')

Out[37]: Text(0.5, 1.0, 'Frequencies of meeting referred objects in each category')
```

Figure 5.4: **Testing whole modules**

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

The proposed system system is an 'opinion mining' system that is used to detect the sentiment in the comments. Through the sentiment detected,it concludes about the review of the topic posted. This kind of system does not demand the human potential. To find about the user's review about any topic,we can use this system which just require the users to post comments about that particular posted topic. As this is an automated operation,this can be considered as an efficient one.

6.2 Comparison of Existing and Proposed System

The existing system is a system in which the users need to check each and every comment posted by the users to know about their opinion on the posted topic. Whereas in the proposed system,we just need the comments of the users. The system automatically detects the positive and negative comments and provide a review about it. In this way, the existing system is a time taking one and the proposed system is a fast process. So,the proposed system can be considered as the more efficient one than the existing system.

Advantages of proposed system

- Time efficient.
- Can avoid reading all the comments posted by users.
- Greater ability to act on customer's suggestions.
- More accurate.

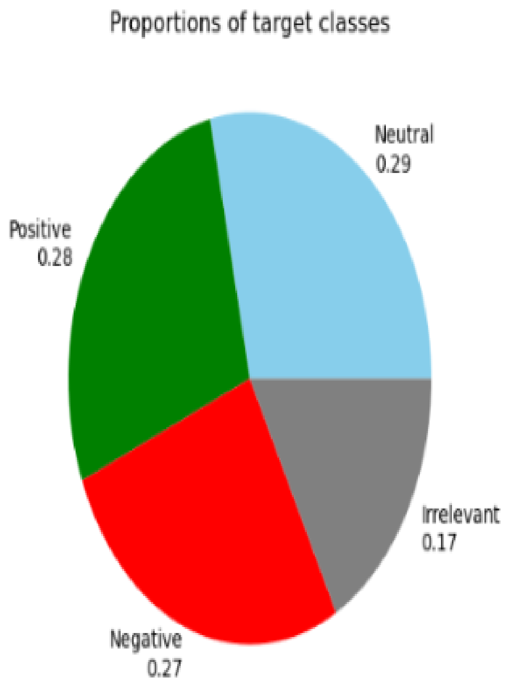
6.3 Sample Code

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
        import pandas as pd
df1=pd.read_csv("C:/Users/mm/Downloads/twitter_validation.csv")
df2=pd.read_csv("C:/Users/mm/Downloads/twitter_training.csv")
df1.head()
df2.head()
plt.figure(figsize=(5, 5))
plt.pie(target_balance, labels=[f'{i}\n{round(target_balance[i]/len(df1), 2)}' for i in
    target_balance.index],
        colors=['skyblue', 'g', 'red', 'grey'])
plt.title('Proportions of target classes')
plt.show()
import re
from nltk.corpus import stopwords
stopwords_list = stopwords.words('english')

word_counts = {'Positive': [],
               'Neutral': [],
               'Irrelevant': [],
               'Negative': []}
  for t, emojis in target_emojis.items():
    plt.figure(figsize=(20, 15))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10, f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.boxplot(x=bar_info.values, y=bar_info.index)
    plt.title(f'{t}')
    plt.show()
```

Output

```
In [24]: for i in outliers.index:
             print(i, 'Target', df1[TARGET][i])
             print(outliers[i])
             print('=-=-=-=-=-=-=-=-=-'*4, '\n')
```

```
In [25]: target_balance = df1[TARGET].value_counts()

         plt.figure(figsize=(5, 5))
         plt.pie(target_balance, labels=[f'{i}\n{round(target_balance[i]/len(df1), 2)}' for i in target_balance.index],
                 colors=['skyblue', 'g', 'red', 'grey'])
         plt.title('Proportions of target classes')
         plt.show()
```

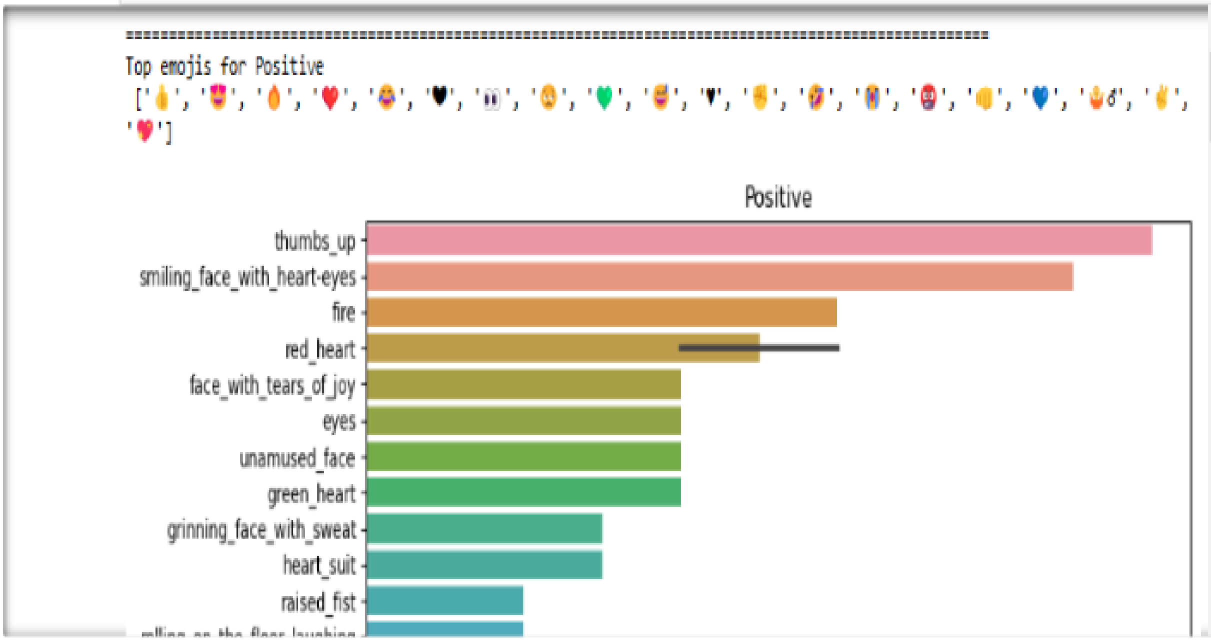



```
In [26]: import re
         from nltk.corpus import stopwords
         stopwords_list = stopwords.words('english')

         word_counts = {'Positive': [],
                        'Neutral': [],
                        'Irrelevant': [],
                        'Negative': []}
```

Figure 6.1: **Piechart visualization**

In [35]:
```python
target_emojis = {'Positive': [],
                 'Neutral': [],
                 'Irrelevant': [],
                 'Negative': []}

pattern = re.compile('\u200d')
for i, text in enumerate(texts):
    emoji_count = split_count(text)
    if emoji_count:
        emoji_count = [re.sub(pattern, '', e) for e in emoji_count]
        target_emojis[df1[TARGET].iloc[i]].extend(emoji_count)
```

In [36]:
```python
for t, emojis in target_emojis.items():
    plt.figure(figsize=(10, 5))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10,  f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.barplot(x=bar_info.values, y=bar_info.index)

    plt.title(f'{t}')
    plt.show()
```



In [37]:
```python
plt.figure(figsize=(8, 10))
sns.heatmap(pd.crosstab(df1['refers to'], df1[TARGET], normalize='index'), annot=True)
plt.title('Frequencies of meeting referred objects in each category')
```

Figure 6.2: **Barchart presentation**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

The sentiment analysis has been a significant endeavor in harnessing the power of natural language processing and machine learning to extract meaningful insights from social media data. Through meticulous data collection, thorough preprocessing, and effective feature extraction, the project has successfully classified sentiments into positive, negative, and neutral categories, providing a nuanced understanding of user opinions.

The insights gained from this project can be applied in diverse areas such as market research, brand sentiment analysis, and social media monitoring, empowering businesses and organizations to make informed decisions based on user feedback. Overall, the sentiment analysis project has showcased the immense potential of leveraging AI technologies to analyze and understand human sentiments expressed in social media, paving the way for future innovations in the field. Overall, the sentiment analysis project has showcased the immense potential of leveraging AI technologies to analyze and understand human sentiments expressed in social media. It has paved the way for future innovations in the field, highlighting the importance of sentiment analysis in extracting valuable insights from the vast amount of data generated on social media platforms.

7.2 Future Enhancements

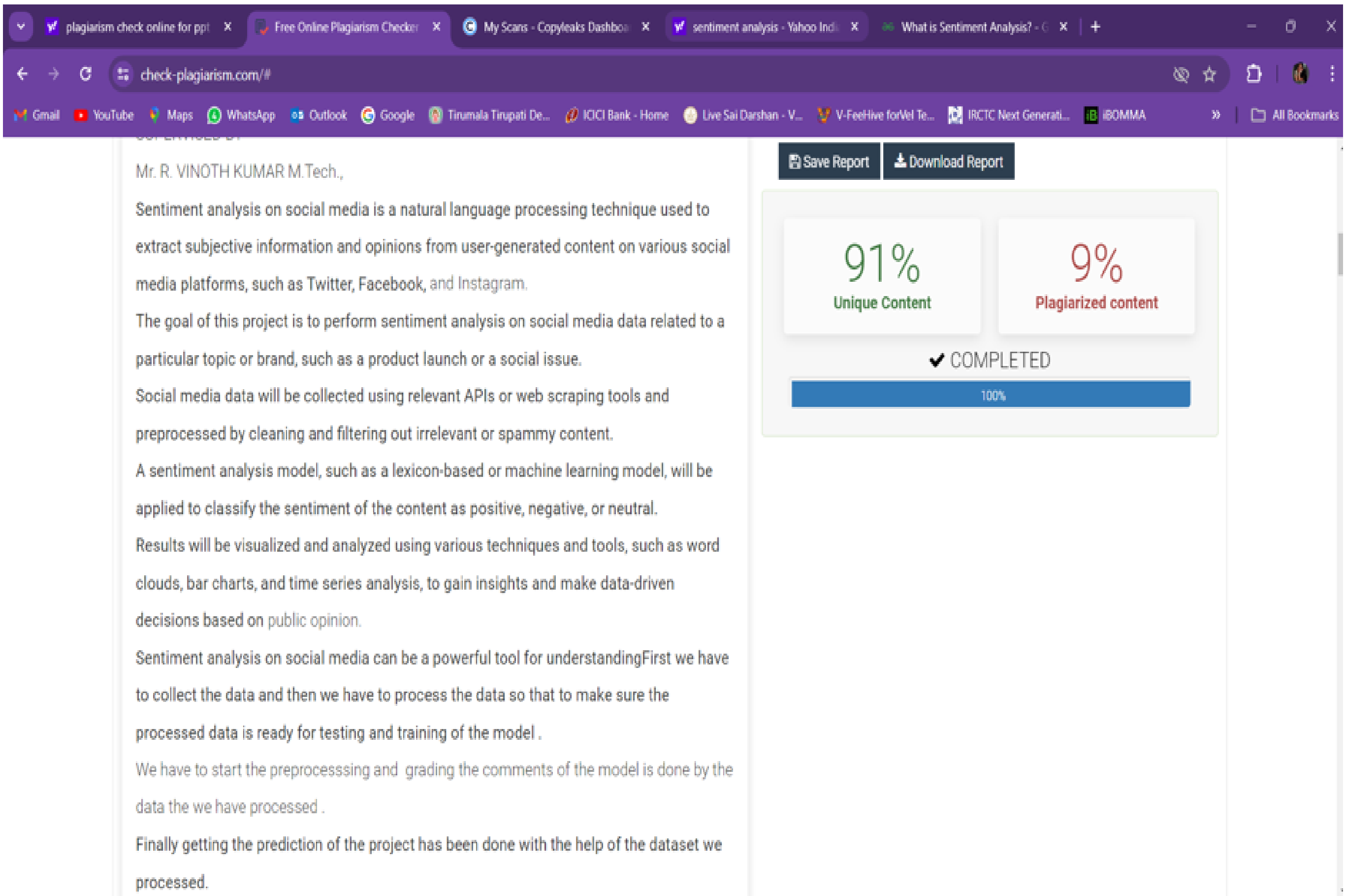
In the future, the sentiment analysis project could be improved by incorporating more advanced techniques, such as deep learning models. These models could help better understand the meaning and context of words in social media posts, leading to more accurate classification of sentiments as positive, negative, or neutral. By leveraging the power of deep learning, the project could potentially uncover more nuanced insights from social media data.

Additionally, integrating real-time data processing technologies could enhance the project's capabilities. Real-time processing would enable the analysis to be faster and more responsive to changes in social media trends. This would allow businesses and organizations to stay up-to-date with the latest developments and sentiments on social media, enabling them to make more informed decisions.

By implementing these enhancements, the sentiment analysis project could become more effective in providing valuable insights from social media data. It could help businesses and organizations gain a deeper understanding of user opinions and sentiments, ultimately leading to better decision-making processes.

Chapter 8

PLAGIARISM REPORT



Chapter 9

SOURCE CODE & POSTER PRESENTATION

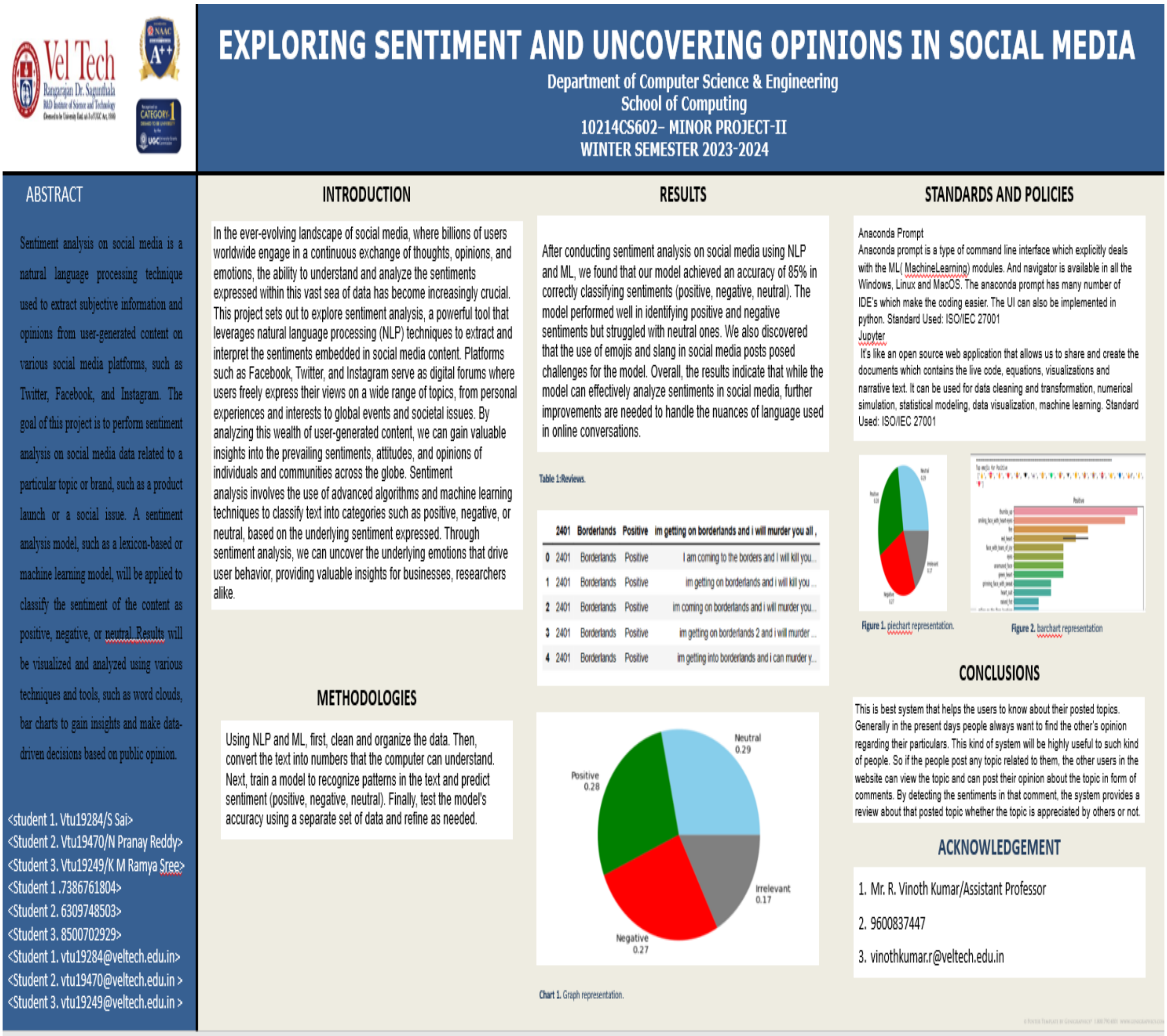
9.1 Source Code

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
        import pandas as pd
df1=pd.read_csv("C:/Users/mm/Downloads/twitter_validation.csv")
df2=pd.read_csv("C:/Users/mm/Downloads/twitter_training.csv")
df1.head()
df2.head()
df1.columns = ['#', 'refers to', 'sentiment', 'text']
df2.columns = df1.columns
df1['#'].value_counts()
df1['refers to'].value_counts()
df1['text'].value_counts()
df1['sentiment'].value_counts()
TARGET = 'sentiment'
import matplotlib.pyplot as plt
import seaborn as sns
df1.info()
df1.isnull().sum()
df2.isnull().sum()
df2.dropna(inplace=True, axis=0)
texts = df1['text']
text_lens = [len(t.split()) for t in texts.values] len_mean = np.mean(text_lens)
fig, axes = plt.subplots(2, 1, figsize=(15, 8)) axes[0].set_title('Distribution of number of tokens
    in tweets') sns.barplot(text_lens, ax=axes[0],color='purple') sns.histplot(text_lens,bins=7, kde
    =True, ax=axes[1],color='r') axes[1].vlines(len_mean, 0, 5000, color = 'g')
plt.annotate("mean", xy=(len_mean, 5000), xytext=(len_mean-2, 5050),color='g')
plt.show()
extreme_outliers = df1['text'][np.array(text_lens) > 125]
for i in extreme_outliers.index:
    print(i, 'Target', df1[TARGET][i])
    print(extreme_outliers[i])
```

```
    print('=-=-=-=-=-=-=-=-=-'*4, '\n')
    outliers = df1['text'][np.array(text_lens) > 60]
    for i in outliers.index:
    print(i, 'Target', df1[TARGET][i])
    print(outliers[i])
    print('=-=-=-=-=-=-=-=-=-'*4, '\n')
    target_balance = df1[TARGET].value_counts()

plt.figure(figsize=(5, 5))
plt.pie(target_balance, labels=[f'{i}\n{round(target_balance[i]/len(df1), 2)}' for i in
    target_balance.index],
        colors=['skyblue', 'g', 'red', 'grey'])
plt.title('Proportions of target classes')
plt.show()
import re
from nltk.corpus import stopwords
stopwords_list = stopwords.words('english')

word_counts = {'Positive': [],
               'Neutral': [],
               'Irrelevant': [],
               'Negative': []}

pattern = re.compile('[^\w ]')
for text, t in zip(df1['text'], df1[TARGET]):
    text = re.sub(pattern, '', text).lower().split()
    text = [word for word in text if word not in stopwords_list]
    word_counts[t].extend(text)
    fig, axes = plt.subplots(2, 2, figsize=(20,10.5))
for axis, (target, words) in zip(axes.flatten(), word_counts.items()):
    bar_info = pd.Series(words).value_counts()[:25]
    sns.histplot(x=bar_info.values, y=bar_info.index, ax=axis)
    axis.set_title(f'Main words for {target}')
plt.show()
fig, axes = plt.subplots(2, figsize=(20,10.5))
for axis, (target, words) in zip(axes.flatten(), word_counts.items()):
    bar_info = pd.Series(words).value_counts()[:25]
    sns.barplot(x=bar_info.values, y=bar_info.index, ax=axis)
    axis.set_title(f'Main words for {target}')
plt.show()
fig, axes = plt.subplots(2,2, figsize=(20,10.5))
for axis, (target, words) in zip(axes.flatten(), word_counts.items()):
    bar_info = pd.Series(words).value_counts()[:25]
    sns.scatterplot(x=bar_info.values, y=bar_info.index, ax=axis)
    axis.set_title(f'Main words for {target}')
plt.show()
tweets_len = {'Positive': [],
               'Neutral': [],
               'Irrelevant': [],
               'Negative': []}
```

```
pattern = re.compile('[^\w ]')
tweets_len = pd.DataFrame([len(re.sub(pattern, '', text).lower().split()) for text in df1['text'] if
    len(text)< 125],
                    columns=['len'])
tweets_len['target'] = df1[TARGET]
plt.figure(figsize=(18, 8))
sns.kdeplot(data=tweets_len, x='len', hue='target')
plt.show()
pip install emoji --upgrade
import emoji
import regex as re
def split_count(text):
    emoji_list = []
    data = re.findall(r'\X', text)
    for word in data:
        if any(char in emoji.EMOJI_DATA for char in word):
            emoji_list.append(word)
     return emoji_list
     target_emojis = {'Positive': [],
                 'Neutral': [],
                 'Irrelevant': [],
                 'Negative': []}

pattern = re.compile('\u200d')
for i, text in enumerate(texts):
    emoji_count = split_count(text)
    if emoji_count:
        emoji_count = [re.sub(pattern, '', e) for e in emoji_count]
        target_emojis[df1[TARGET].iloc[i]].extend(emoji_count)
for t, emojis in target_emojis.items():
    plt.figure(figsize=(10, 5))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10, f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.barplot(x=bar_info.values, y=bar_info.index)
    plt.title(f'{t}')
    plt.show()
plt.figure(figsize=(8, 10))
sns.heatmap(pd.crosstab(df1['refers to'], df1[TARGET], normalize='index'), annot=True)
plt.title('Frequencies of meeting referred objects in each category'
for t, emojis in target_emojis.items():
    plt.figure(figsize=(20, 15))
    bar_info = pd.Series(emojis).value_counts()[:20]
    print('=========='*10, f'\nTop emojis for {t} \n', list(bar_info.index))
    bar_info.index = [emoji.demojize(i, delimiters=("", "")) for i in bar_info.index]
    sns.boxplot(x=bar_info.values, y=bar_info.index)
    plt.title(f'{t}')
    plt.show()
```

9.2 Poster Presentation



References

1. Y. Fang, H. Tan and J. Zhang, ”Multi-Strategy Sentiment Analysis of Consumer Reviews Based on Semantic Fuzziness,” IEEE Access, vol. 6, pp. 20625-20631, 2018.
2. M. Afzaal, M. Usman and A. Fong, ”Tourism Mobile App With Aspect-Based Sentiment Classification Framework for Tourist Reviews,” IEEE Transactions on Consumer Electronics, vol. 65, no. 2, pp. 233-242, May 2019.
3. 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,” IEEE Access, vol. 7, pp. 83354-83362, 2019.
4. NeelamMukhtar, Mohammad AbidKhan, and NadiaChiragh, ”Lexicon-based approach outperforms Supervised Machine Learning approach for Urdu Sentiment Analysis in multiple domains”, Telematics and Informatics, vol. 35, no. 8, pp. 2173-2183, December 2018.
5. AkshiKumar, KathiravanSrinivasan, ChengWen-Huang, and Albert Y.Zomaya, ”Hybrid context enriched deep learning model for fine-grained sentiment analysis in textual and visual semiotic modality social data”, Information Processing Management, vol. 57, no. 1, January 2020.
6. AsadAbdi, Siti MariyamShamsuddin, ShafaatunnurHasan, and JalilPiranMD, ”Machine learning-based multi-documents sentiment-oriented summarization using linguistic treatment”, Expert Systems with Applications, vol. 109, pp. 66-85, 1 November 2018.
7. ParamitaRay, and AmlanChakrabarti, ”A Mixed approach of Deep Learning method and Rule-Based method to improve Aspect Level Sentiment Analysis”, Applied Computing and Informatics, Available online 4 March 2019.
8. SrishtiVashishtha, and SebaSusan, ”Fuzzy rule based unsupervised sentiment analysis from social media posts”, Expert Systems with Applications, vol. 138, 30 December 2019.
9. AbdallahYousif, ZhendongNiu, JamesChambua, and Zahid YounasKhan, ”Multi-task learning model based on recurrent convolutional neural networks

for citation sentiment and purpose classification”, Neurocomputing, vol. 335, pp. 195-205, 28 March 2019.

10. Mohammad A.Hassonah, RizikAl-Sayyed, AliRodan, Ala’ M.Al-Zoubi, IbrahimAljarah, and HossamFaris, ”An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter”, Knowledge-Based Systems, vol. 192, 15 March 2020.
11. FengXu, ZhenchunPan, and RuiXia, ”E-commerce product review sentiment classification based on a naïve Bayes continuous learning framework”, Information Processing Management, Available online 13 February 2020.
12. MohammadAl-Smadi, OmarQawasmeh, MahmoudAl-Ayyoub, YaserJararweh, and BrijGupta, ”Deep Recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels’ reviews”, Journal of Computational Science, vol. 27, pp. 386-393, July 2018.