OPINION MINING AND SENTIMENT ANALYSIS ON SOCIAL MEDIA

Minor project-1 report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science and Design

By

NANDYALA NAVEEN REDDY	(21UEDL0020)	**(VTU20906)**
DALLI AKASH REDDY	(21UEDL0008)	**(VTU20933)**
G SIVA GANGADHAR	(21UECE0022)	**(VTU20690)**

Under the guidance of
Dr. A. BHAGYALAKSHMI, M.E., Ph.D.,
PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE AND DESIGN
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

January, 2024

OPINION MINING AND SENTIMENT ANALYSIS ON SOCIAL MEDIA

Minor project-1 report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science and Design

By

NANDYALA NAVEEN REDDY	(21UEDL0020)	**(VTU20906)**
DALLI AKASH REDDY	(21UEDL0008)	**(VTU20933)**
G SIVA GANGADHAR	(21UECE0022)	**(VTU20690)**

Under the guidance of
Dr. A. BHAGYALAKSHMI, M.E., Ph.D.,
PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE AND DESIGN
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

January, 2024

CERTIFICATE

It is certified that the work contained in the project report titled "OPINION MINING AND SENTIMENT ANALYSIS ON SOCIAL MEDIA" by "NANDYALA NAVEEN REDDY (21UEDL0020), DALLI AKASH REDDY (21UEDL0008), G SIVA (21UECE0022)" has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

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

Signature of Head of the Department
Computer Science and Design
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
January, 2024

Signature of the Dean
Dr. V. Srinivasa Rao
Professor & Dean
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
January, 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.

NANDYALA NAVEEN REDDY

Date: / /

DALLI AKASH REDDY

Date: / /

G SIVA GANGADHAR

Date: / /

APPROVAL SHEET

This project report entitled OPINION MINING AND SENTIMENT ANALYSIS ON SOCIAL MEDIA by N NAVEEN REDDY (21UEDL0020), D AKASH REDDY (21UEDL0008), G SIVAGANGADHAR (21UECE0022) are approved for the degree of B.Tech in Computer Science and Design.

Examiners **Supervisor**

Dr. A. Bhagya Lakshmi, M.E., Ph.D.,

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 and Design, Dr. R. PARTHASARATHY, 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 **Dr. A. BHAGYALAKSHMI, M.E., Ph.D.,** for her cordial support, valuable information and guidance, she helped us in completing this project through various stages.

A special thanks to our **Project Coordinators Mr. V. ASHOK KUMAR, M.Tech., Ms. C. SHYAMALA KUMARI, M.E., Mr. SHARAD SHANDHI RAVI, M.Tech.,** 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.

NANDYALA NAVEEN REDDY (21UEDL0020)

DALLI AKASH REDDY (21UEDL0008)

G SIVA GANGADHAR (21UECE0022)

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 preprocessed by cleaning and filtering out irrelevant or spammy 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 : Filtering, Machine learning, Decision Tree Natural Language Processing(NLP), Sentiment, Socialmedia, Visual Studio Code, Jupyter.

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

API	Application Programming Interface
CSS	Cascading Style Sheet
HTML	Hyper Text Markup Language
ML	Machine Learning
NLP	Natural Language Processing
UI	User Interface
VS	Visual Studio

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Sentiment analysis is a natural language processing and machine learning technique used to extract subjective information from user-generated content on social media platforms. Social media sentiment analysis can help companies and organizations to monitor brand reputation, identify emerging trends and patterns, and make data-driven decisions based on public opinion. To start a sentiment analysis project, we need to define the scope and goals of project, choose a social media platform and data type, and collect and pre process the data. Pre processing involves cleaning and preparing the data, such as removing irrelevant or spammy content, removing stop words, and converting text to lowercase. There are various sentiment analysis models and techniques, such as rule-based models, lexicon-based models, and machine learning models, that can be applied to the pre processed data to classify sentiment as positive, negative, or neutral. After sentiment classification, the results can 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. Sentiment analysis on social media can be a powerful tool for understanding public opinion and making data-driven decisions in various domains, such as marketing, politics, and social sciences.

This is a system that allows the users to take decision about their topics/products shared in the app. The users can post their topics to get a review about it. This review is collected based on the comments of users. This system actually grades a mark for each comment by comparing the sentiment in the comment with it's dataset. By comparing all the grades, this system returns a review about the posted topic. As the title defines this is just a method to collect the user's opinion in form of comments and provide a review to the posted topic. The users in this site can even edit their profiles.By using this kind of system the user can easily come to know whether his topic is an appreciable one or not.

Social media platforms generate a vast amount of unstructured data, including user-generated content, such as text, images, and videos. Sentiment analysis techniques enable businesses to analyze this data to identify and classify the emotional tone of social media posts.The sentiment analysis process involves several steps, including data collection, preprocessing, feature extraction, classification, and evaluation. In the data collection step, social media posts are collected using various tools and techniques, such as web scraping or API calls. In the preprocessing step, the data is cleaned, normalized, and tokenized to remove irrelevant information and standardize the text format. This step is essential for ensuring that the data is consistent and uniform for analysis.

The feature extraction step involves converting the text data into numerical features, such as word frequencies or vector embeddings, to enable the application of machine learning algorithms for classification. Various classification algorithms, such as Naive Bayes, Support Vector Machines(SVM), or Deep Learning models(DL), can be used to predict the sentiment of the social media post. The classification algorithm learns from a labeled dataset of positive, negative, and neutral sentiments and predicts the sentiment of new, unlabeled data.Finally, the evaluation step is used to measure the accuracy of the sentiment analysis model by comparing the predicted sentiment with the actual sentiment. This step helps to refine the model and improve its accuracy over time.

Sentiment analysis on social media can provide businesses with a wealth of information about their customers. It can help them understand how their customers feel about their products or services, what they like or dislike, and what they are saying about them. By analyzing social media sentiment, businesses can monitor their brand reputation and respond promptly to negative feedback or complaints. They can also use this information to identify trends and patterns in customer sentiment and make data-driven decisions to improve their products or services. It is a powerful tool for businesses to gain insights into customer sentiment and improve their overall customer experience. It enables businesses to monitor their reputation, track customer feedback, and identify opportunities for improvement. By leveraging sentiment analysis techniques, businesses can stay ahead of their competition and build a strong relationship with their customers.

1.2 Aim of the Project

The main aim of sentimental analysis on social media is to automatically classify and analyze the opinions, sentiments, and emotions expressed by users on social media platforms. To develop a software that decides whether the posted topic is a usable one/non usable one based on the comments of the users. This analysis can provide valuable insights into the attitudes and opinions of users towards different topics, products, or services.

1.3 Project Domain

The domain used to implement this project is 'Machine Learning'. Machine learning is perhaps the principal technology behind two emerging domains: data science and artificial intelligence. The rise of machine learning is coming about through the availability of data and computation, but machine learning methodologies are fundamentally dependent on models. The emergence of machine learning is closely tied to the emergence of widely available data. In this domain we use Naive Bayes algorithm for prediction.

1.4 Scope of the Project

The scope of the project is to judge whether our product is a good one/bad one. This helps the people to know the opinion of others regarding any posted one. The developers can understand their faults and can correct their mistakes. This makes to enhance the quality and popularity of the product. This helps to find the harmful content on social media platform based on the reviews given by the users.

Chapter 2

LITERATURE REVIEW

[1]Thai Kim Phung et al, proposed that this study was conducted on a theoretical background of opinion mining methods, opinion classification techniques and proposed the application of supervised machine learning method for automatic opinion mining. Experimental results show that LR, SVM and NN are the best among the training methods. This study is valuable as a reference for applications of opinion mining in socio-economy field.

[2]Fahad Alhujaili et al, addressed that the sentiment analysis is a topic under natural language processing. It concerns on analysis a user opinion against any product or service. Such way can help in improving the products, service or help in change path of any bad service or product. This paper provided a comprehensive review for sentiment analysis methods based on YouTube videos. These methods into three levels: simple SA, complex SA, advanced SA. This paper can help many researchers in research area of sentiment analysis or related areas.

[3]Nourin Islam et al, proposed that the approach targets reacting to inquiries concerning the drivers of each stamped notion in a dataset and examine the overall broadness of the idea applied regular ML grouping approaches for finding the supposition investigation for food audit dataset. SVM give great precision for estimation examination.

[4]Nikhil Yadav et al, explained that comparing different classifiers to classify a huge number of English tweets related to particular products into positive as well as negative sentiments. Using sentiment features rather than conventional text classification gives high accuracy.

[5]Amandeep Kaur et al, developed that as it can be applied in many fields Opinion mining is an interesting area in the field of research. This kind of collecting the opinion of users about any product and services is becoming huge day by day. At the time of making decisions, this kind of reviews are much more helpful. The opinion mining is a way to collect/gather information through search engines, social network. Accordingly efficient computational methods are needed for mining and summarizing the reviews.

[6]P.Balaji et al, explained that the availability of opinion mining resources like online review sites and blogs will make the people to know the user's review and experience towards any product. These websites make the users to provide the review and rating to any product and services. These review can be of negative and positive based on the user's experience regarding the products. This opinion decides the product's review.

[7]Ms.Binju Saju et al,proposed that sentiment analysis can serve as the solution to several business problems. It is the nexus of several technologies and accumulated knowledge. It is safe to say that sentiment analysis is a multi-disciplinary domain.The overall quality of the information available for decision making is greatly improved. The advancements in machine learning also mean improvement of sentiment analysis techniques.

[8]Sheresh Zahoor et al,addressed the sentiments can be predicted with more accuracy using Machine Learning and Deep Learning algorithmsespecially Naïve Bayes, SVM, Random Forest Classifier and LSTM. These have found their use in the field of Natural Language Processing especially in Sentiment analysis or to determine the subjective information like opinion, emotion within a piece of text. However, the accuracy with which each of them is able to predict the sentiment varies.

[9]H.Soong et al,explained that the concept of opinion mining involve many fields like natural language processing, text mining and decision mining. This is a text analysis that classifies the text and make decisions by analyzing the text. The opinion is classified into 2 types. They are positive opinion and Negative Opinion. The type of the user's opinion decides the review of the product.

[10]Kumar Ravi et al,addressed that in today's world people are getting more attracted to share their reviews and opinions regarding any product and services in the social media. Evolution of social media is majorly contributing to these activities. Hence, offering the users an effective platform to share and spread their reviews throughout the world. The major reason for collecting opinions regarding any product is to know and analyze the user's interest and mood regarding the products.

[11]Arti et al,explained that opinions for sporting game such as Twitters API give access to authorized users to extract the details insights into tweets post. The practical result displays the positive think and negative think of individuals respectively. This type of an opinion analysis might offer valuable feedback to the business and facilitate them to identify a negative tweets flip in viewer's understanding. Deciding negative trends too soon will permit them to form educated choices on a way to target specific aspects of their services and products so as to extend its client satisfaction.

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. 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. 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

Opinion Mining is a process used to extract the user's opinion. The technique involved in this process is Sentiment Analysis. The comments posted by the users will be seen by the system and recognize the sentiment in the comment. Based on the sentiment it decides whether the posted topic is an appreciable one or not. The sentiment in the comments can be positive or negative. If the extracted sentiment is a positive one, we can conclude that the posted topic is a good or nice one and if the sentiment extracted is a negative one, the posted topic is not a good one. So this kind of operation does not consume much time. This kind of projects are highly useful to the people who are going to discover or invent new things. By posting new thing they can know about the positives and negatives before discovering itself. It can be used as an advertisement also.

Advantages:

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

3.3 Feasibility Study

As, the proposed project is a simple and easy one, we can consider that as a simple model. It may take a little time to develop this model. It is a good one and highly useful in the society.

3.3.1 Economic Feasibility

The model 'Opinion Mining' based on text analysis in online social network's does not demand any rear and costlier resources. It is a very simple and easy project. As it is a software model, there is no need to spend money. It can be considered as an economically feasible.

3.3.2 Technical Feasibility

This model definitely require a basic knowledge in technical field. Also, much depth in the technical field is also not required. As it is mostly based on simple machine learning algorithms using python language, it can be easier to develop for those who has basic knowledge in this domain. We can consider that it as a technically feasible one.

3.3.3 Social Feasibility

In the present days this kind of model is highly required by the people. This kind of model is highly applicable on social media platforms and people in the present days are highly active in social media. This is definitely a social feasible.

3.4 System Specification

3.4.1 Hardware Specification

- Processor : i5
- Hard Disk : 5 GB
- ROM : 1TB
- Keyboard : Standard Windows Keyboard
- Mouse : A Standard Mouse
- Memory : 1GB RAM

3.4.2 Software Specification

- Windows
- Visual Studios 2010

3.4.3 Standards and Policies

Anaconda Prompt

Anaconda prompt is a type of command line interface which explicitly deals with the ML(MachineLearning) modules.And navigator is available in all the Windows,Linux and MacOS.The anaconda prompt has many number of IDE's which make the coding easier. The UI can also be implemented in python.

Standard Used: ISO/IEC 27001

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.

Standard Used: ISO/IEC 27001

Chapter 4

METHODOLOGY

4.1 General Architecture

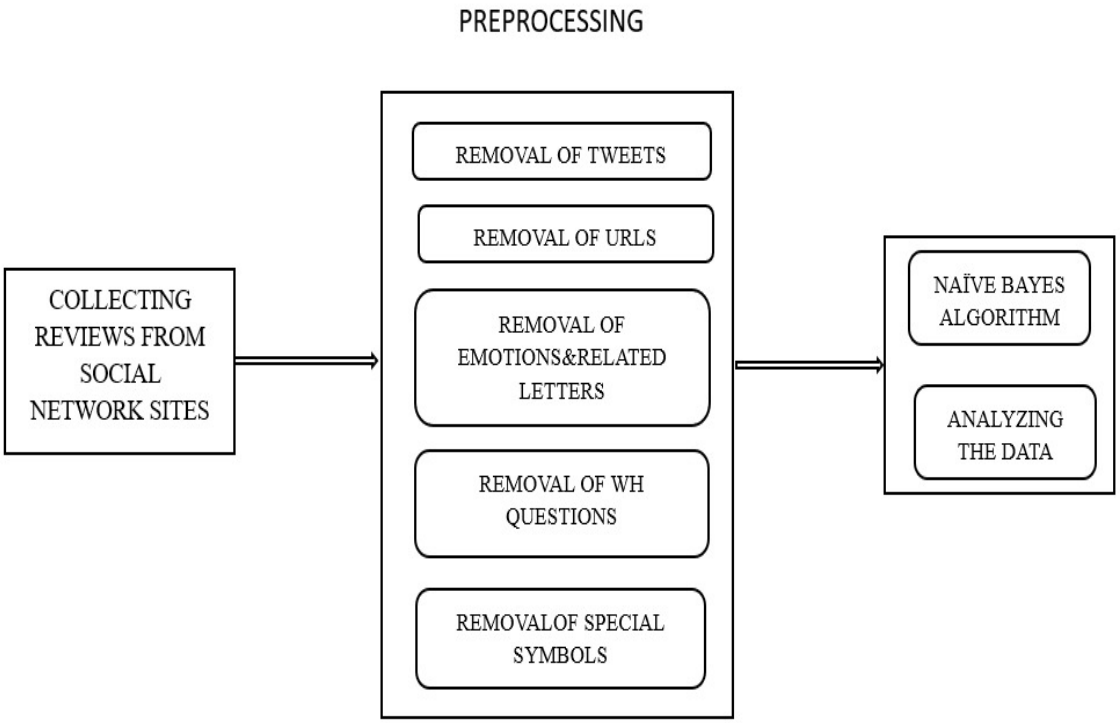


Figure 4.1: **Architecture Diagram**

The Figure 4.1 represents 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- words are: what, when, where, who, whom, which, whose, why, and how. Then last step in the data preprocessing 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

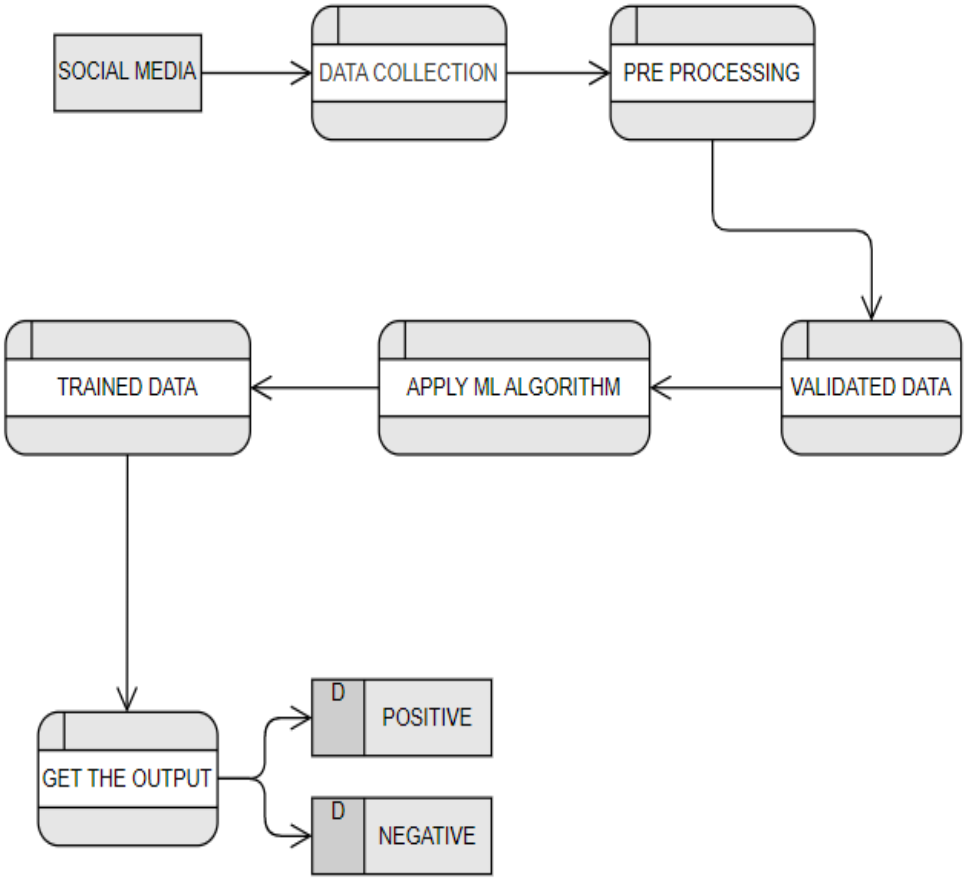


Figure 4.2: **Data Flow Diagram**

The Figure 4.2 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. 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

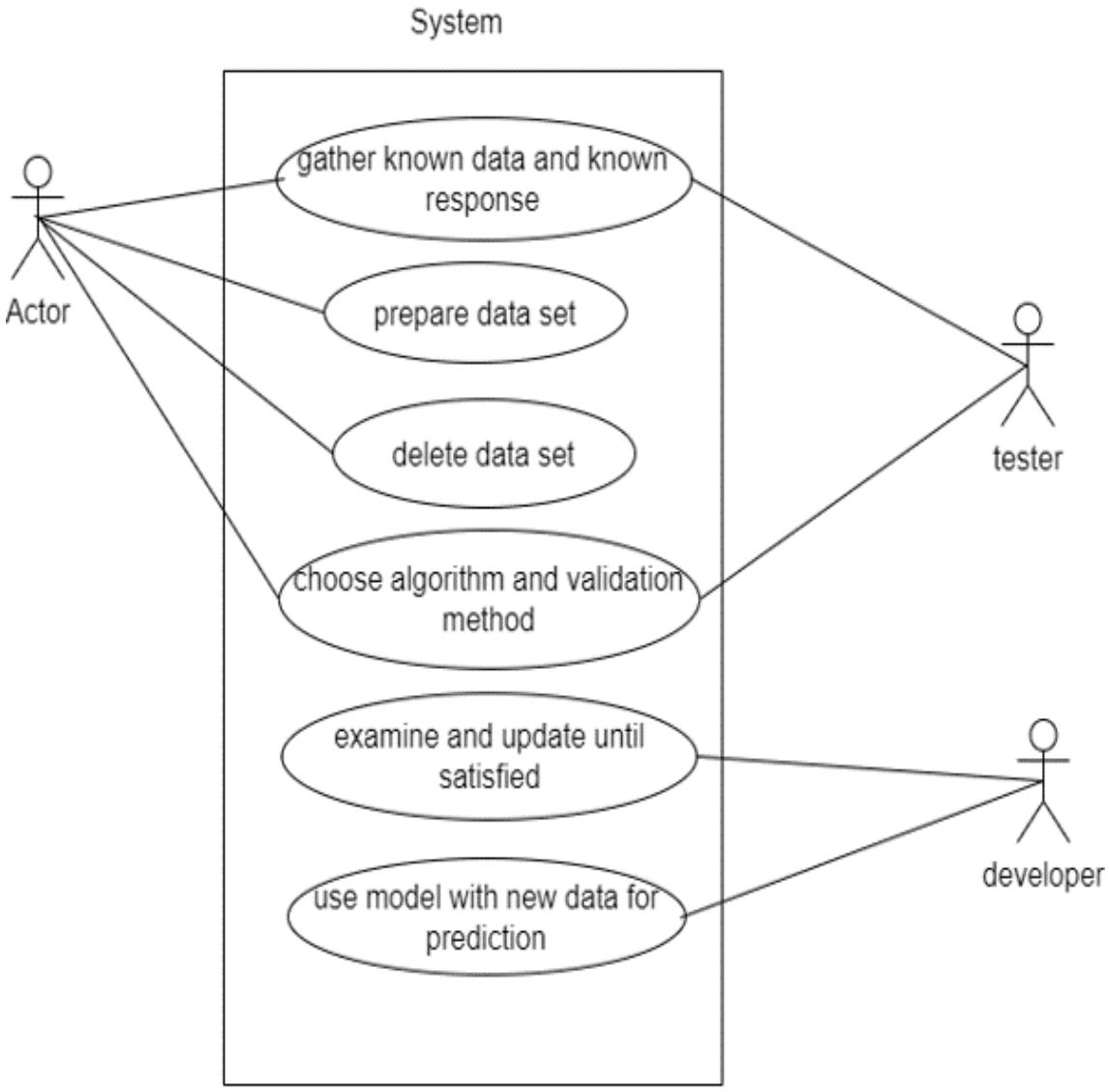


Figure 4.3: **Use Case Diagram**

The Figure 4.3 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. 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

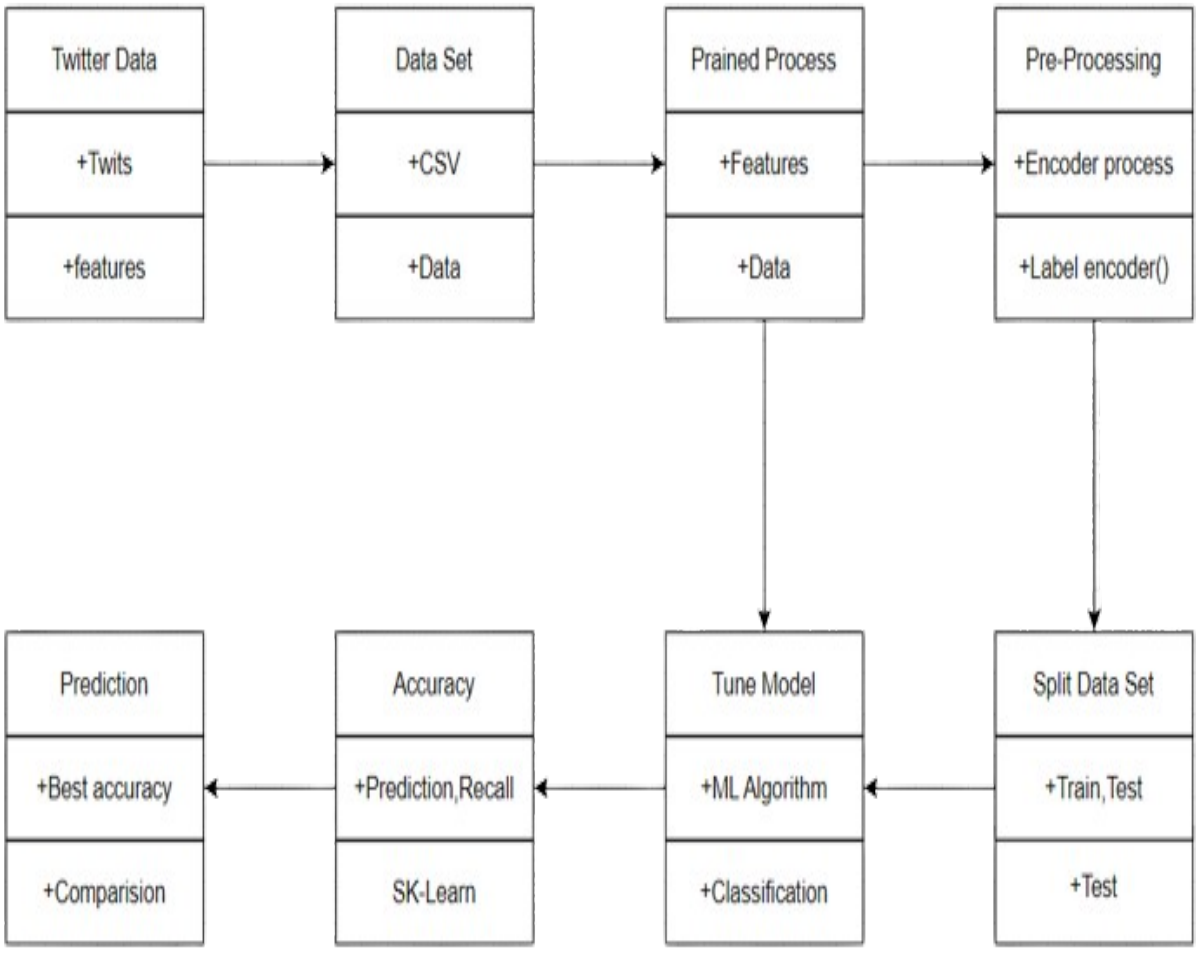


Figure 4.4: **Class Diagram**

This Figure 4.4 depicts the different classes in opinion mining process. Following components; Twitter data: This class the data of twitts, features. Data set: This class contain the csv and data of twitts stored in the csv file. Trained Process : This class 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

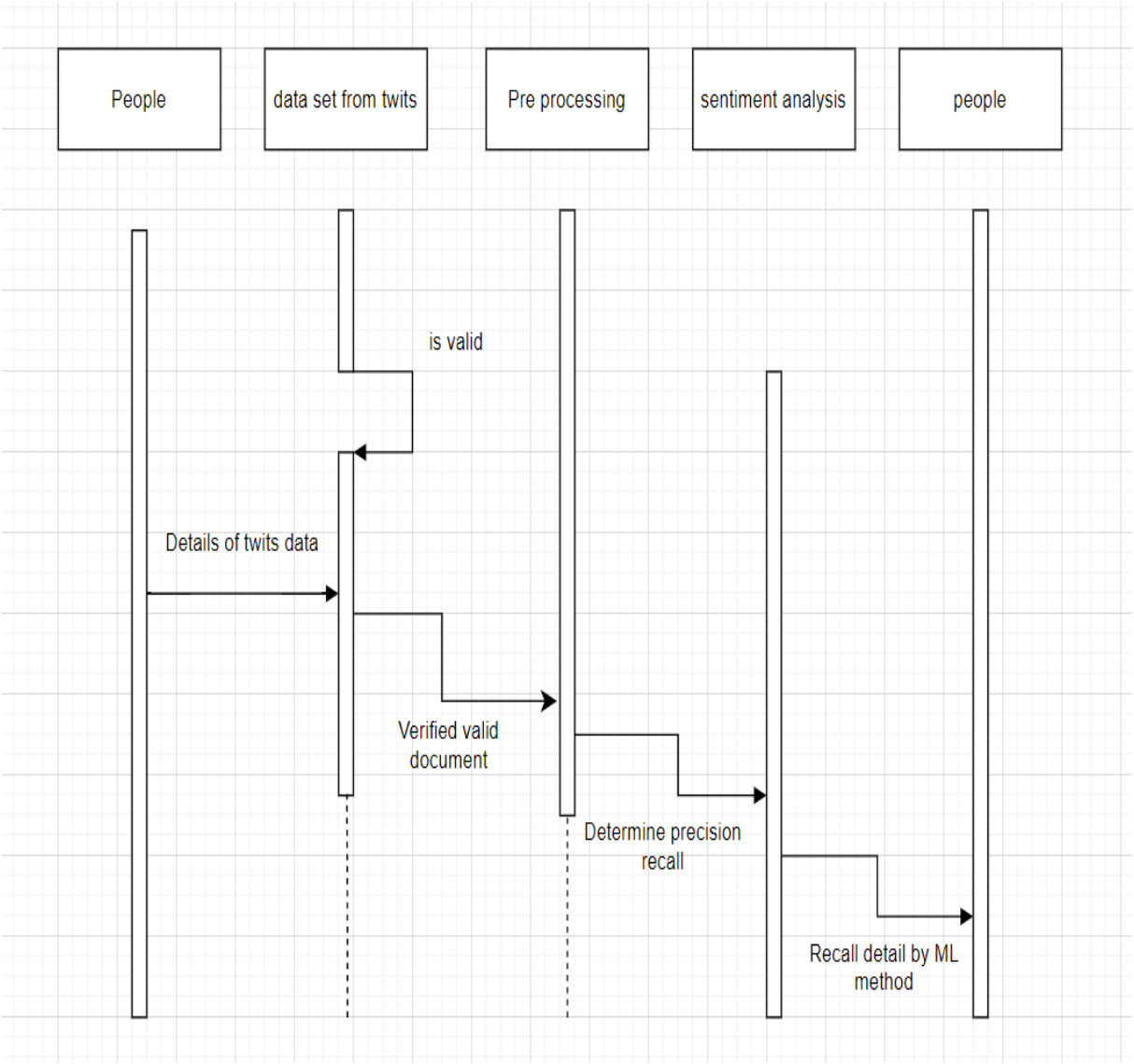


Figure 4.5: **Sequence Diagram**

The Figure 4.5 user first logs in to the social network. Then he sees the posted topic and posts his opinion in form of comments. 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. Then the classification will be done and it provides a review about the posted topic.

4.2.5 Activity Diagram

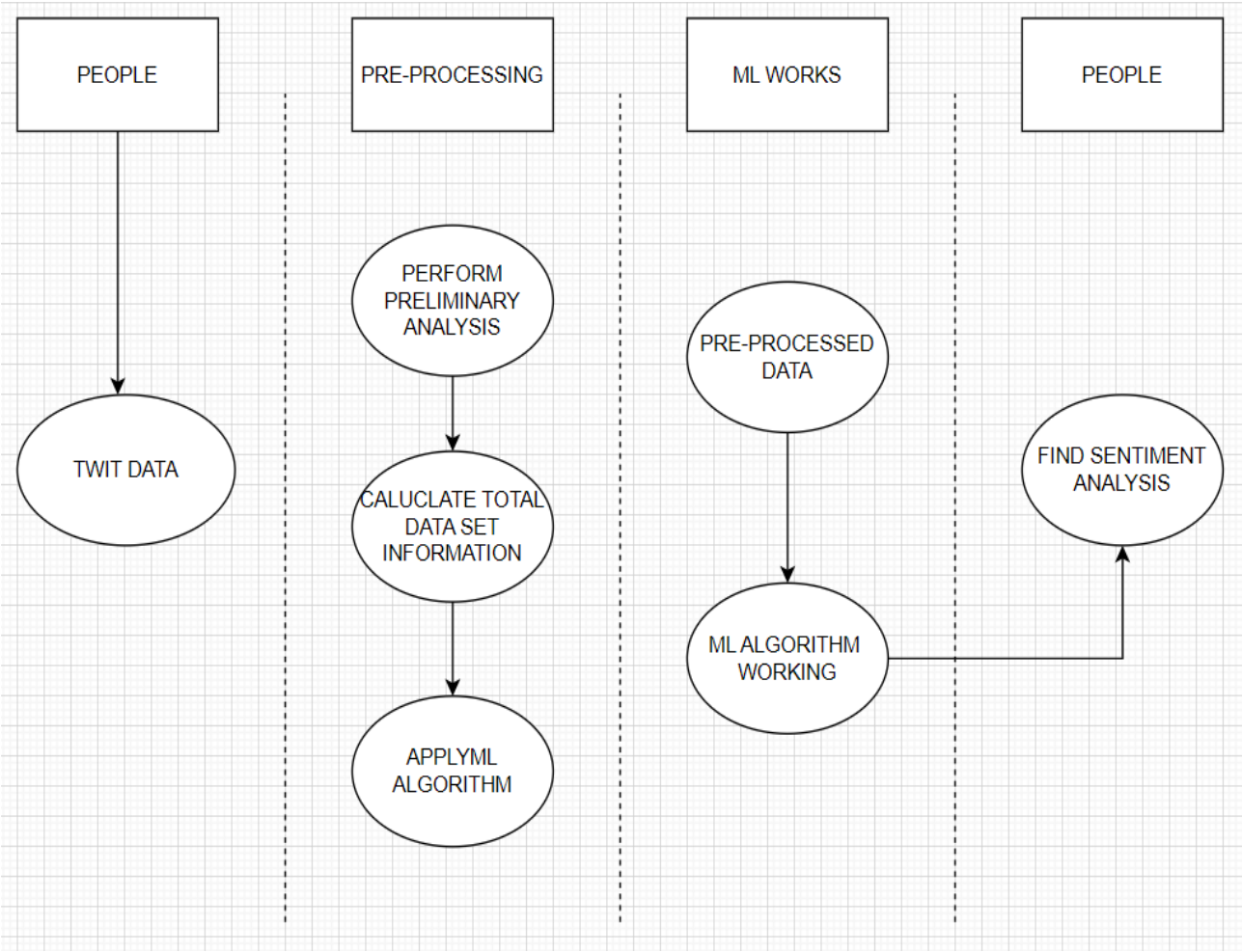


Figure 4.6: **Activity Diagram**

The Figure 4.6 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 Algorithm

1. Import all required library files.
2. Get the data file to read the reviews.
3. Grade the reviews based on repeated words.
4. Separate the positive and negative,irrelevant words.
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.

4.3.2 Pseudo Code

```
import numpy as np
import pandas as p
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import re
from nltk.corpus import stopwords
stopwords_list = stopwords.words('english')

word_counts = {'Positive': [],
               'Neutral': [],
               'Irrelevant': [],
               'Negative': []}
               pattern = re.compile('\u200d')
```

4.4 Module Description

4.4.1 Upload Module

It is a module in which the people post any topic related to them. This posting of topic will make the others users to see the topic. The users who view that posted topic can comment their opinion about the posted topic. We download the dataset from Kaggle and extracted the dataset. The data set we download will appear as follows.

4.4.2 Comment Module

This is a module in which the users post their opinions in form of comments about the topic posted by other people. In this module,everyone can see the comments posted by the people. The comment is just like a text message. It allows to type small, big characters, numbers and symbols, emojis.

4.4.3 Classifying Module

It is a module that will be taken care by the system. The system recognizes the comments and based on the sentiment it classifies the negative and positive comments. Based on the classification it provides the grade for each comment.

4.4.4 Grading Module

The Grading Module will be taken care by the system itself. In this module,after classification of comments it grades a mark to each comment. Based on the grade it provided we come to a decision about the review of the topic posted. If the mark provided is a bigger number, we consider that the posted topic is appreciated by the users and if the mark is a smaller number, we consider that, it is not appreciable by the users.

4.5 Steps to execute/run/implement the project

4.5.1 Import libraries

- Importing all required library files.

4.5.2 Reading data

- Declare the path to the file and get it.
- Read the data in the file and the file should be in CSV format.

4.5.3 Training Testing

- Find all the repeated words and unwanted words in data.

- clear all those unwanted data.
- train test the model using the data.

4.5.4 Pre processing

- Using Naive Bayes algorithm predict the probability.
- Calculate and get the sentiment value and display as output.

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()

Out[4]:
```

	3364	Facebook	Irrelevant	I mentioned on Facebook that I was struggling for motivation to go for a run the other day, which has been translated by Tom's great auntie as 'Hayley can't get out of bed' and told to his grandma, who now thinks I'm a lazy, terrible person
0	352	Amazon	Neutral	BBC News - Amazon boss Jeff Bezos rejects clai...
1	8312	Microsoft	Negative	@Microsoft Why do I pay for WORD when it funct...
2	4371	CS-GO	Negative	CSGO matchmaking is so full of closet hacking,...
3	4433	Google	Neutral	Now the President is slapping Americans in the...
4	6273	FIFA	Negative	Hi @EAHelp I've had Madeleine McCann in my cel...

```
In [5]: df2.head()

Out[5]:
```

	2401	Borderlands	Positive	im getting on borderlands and i will murder you all ,
0	2401	Borderlands	Positive	I am coming to the borders and I will kill you...
1	2401	Borderlands	Positive	im getting on borderlands and i will kill you ...
2	2401	Borderlands	Positive	im coming on borderlands and i will murder you...
3	2401	Borderlands	Positive	im getting on borderlands 2 and i will murder ...
4	2401	Borderlands	Positive	im getting into borderlands and i can murder y...

Figure 5.1: **Input Dataset**

5.1.2 Output Design

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

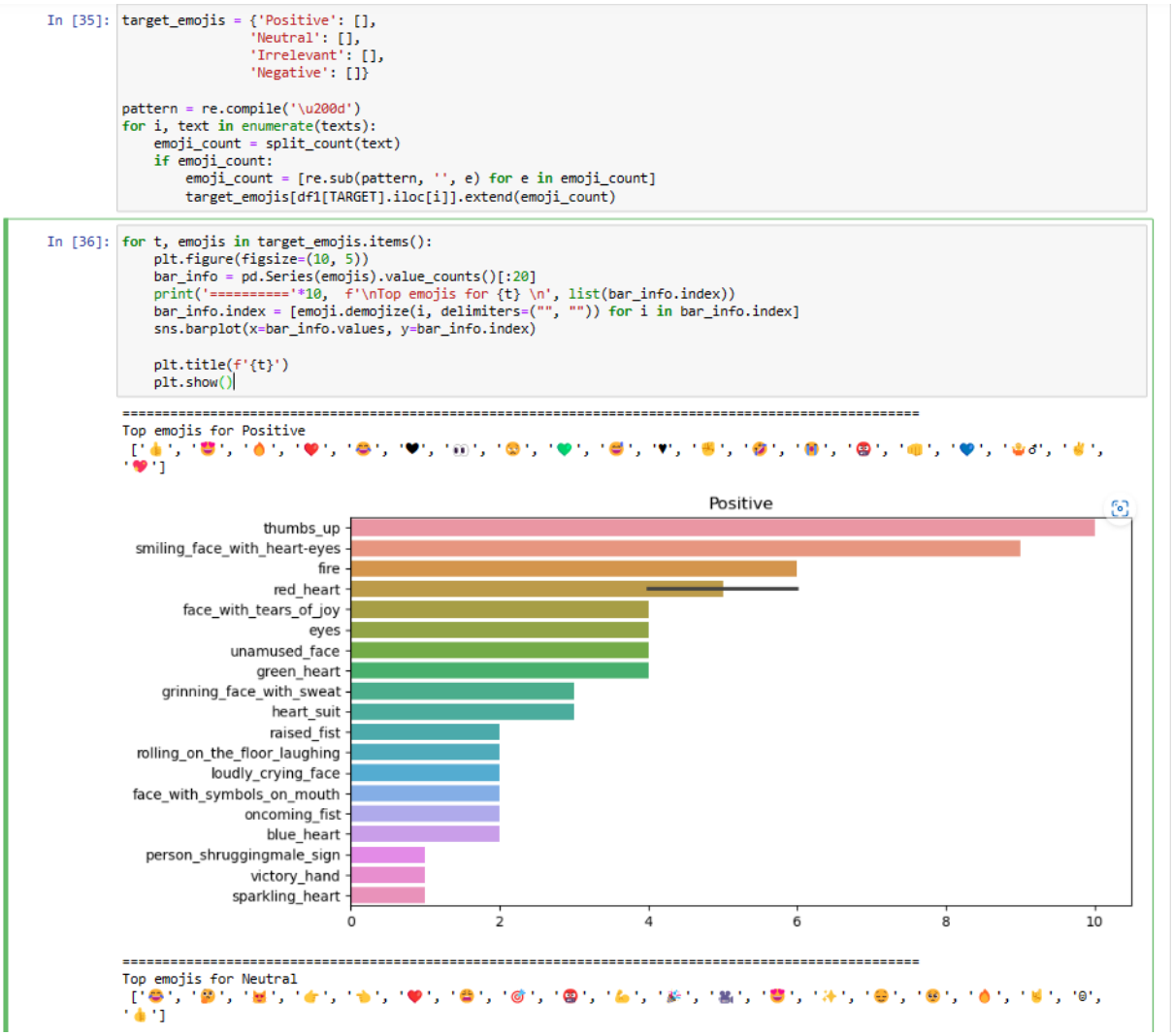


Figure 5.2: **Output Image**

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

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

```
In [7]:  df1['#'].value_counts()

Out[7]: 352     1
        6890    1
        364     1
        2567    1
        7780    1
               ..
        8955    1
        1565    1
        3361    1
        8408    1
        6960    1
        Name: #, Length: 999, dtype: int64

In [8]: df1['refers to'].value_counts()

Out[8]: RedDeadRedemption(RDR)              40
        johnson&johnson                    39
        FIFA                               38
        PlayerUnknownsBattlegrounds(PUBG)  38
        LeagueOfLegends                    37
        ApexLegends                        36
        Nvidia                             35
        TomClancysRainbowSix               35
        GrandTheftAuto(GTA)                35
        Fortnite                           34
        Amazon                             34
        AssassinsCreed                     33
        Borderlands                        33
        PlayStation5(PS5)                  33
        Hearthstone                        32
        Overwatch                          32
        Verizon                            32
        CS-GO                              32
        Facebook                           32
        CallOfDuty                         31
        Cyberpunk2077                      30
        WorldOfCraft                       30
        MaddenNFL                          29
```

Figure 5.3: **Sentiment Count Module**

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

```
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()
```

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()
```

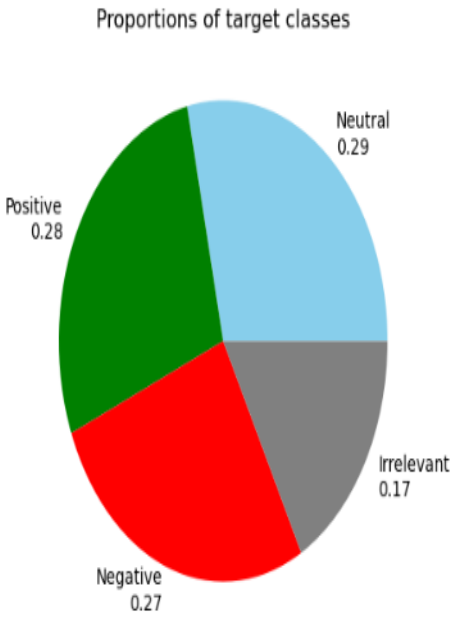


Figure 5.4: **Opinion Bargraph in Target Classes**

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()
```

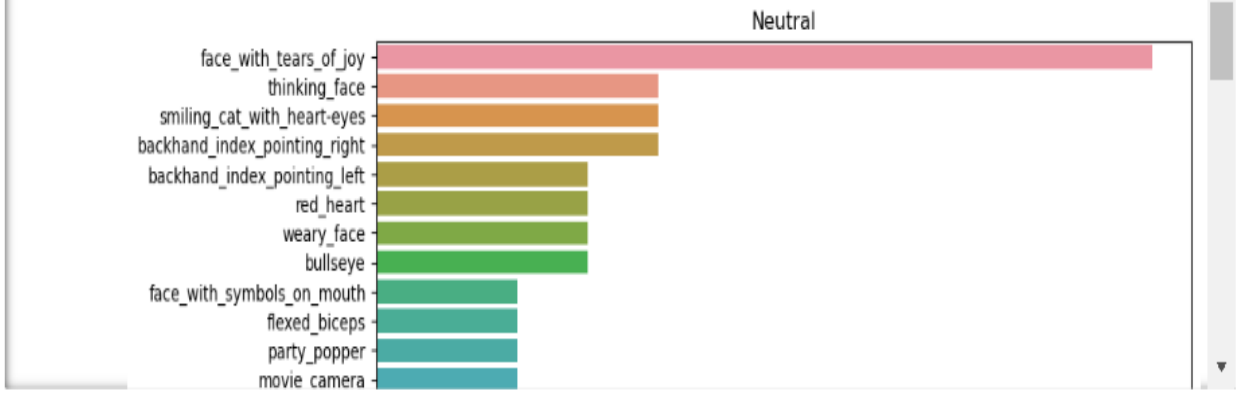
5.3.4 Test Result

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

```
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 Neutral
['😂', '😭', '😻', '👉', '👈', '❤️', '😍', '🎯', '😡', '💪', '🎉', '🎆', '😻', '✨', '😩', '😲', '🔥', '🤘', '🅾', '🖕']
```




```
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 5.5: **Test Result**

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

The proposed 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(opinion mining 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. 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.

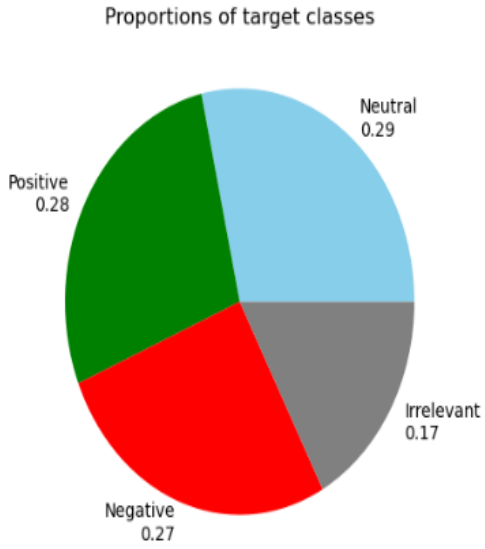
6.3 Sample Code

```
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()
```



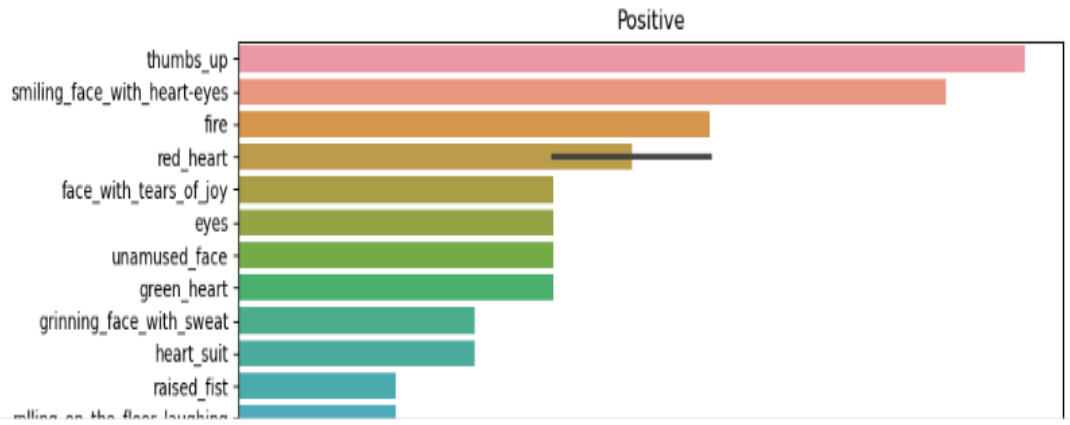

```
In [26]: import re
         from nltk.corpus import stopwords
         stopwords_list = stopwords.words('english')

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

Figure 6.1: **Piechart Visualization**

```
In [35]: 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
 ['🔥', '😍', '🔥', '❤️', '😂', '🖤', '👀', '😭', '💚', '😻', '♥️', '✋', '🤣', '🤗', '🤯', '👊', '💙', '🤷‍♂️', '✌️',
'💔']
```




```
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')
```

Figure 6.2: **Bargraph Representation for Postivie Emoji's**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

The Opinion Mining system is a kind of system that is used to provide a review about any kind of posted topic. This is best system that helps the users to know about their posted topics. Generally in the present days people always want to find the other's opinion regarding their particulars. This kind of system will be highly useful to such kind of people. If the people post any topic related to them, the other users in the website can view the topic and can post their opinion about the topic in form of comments. By detecting the sentiments in that comment, the system provides a review about that posted topic whether the topic is appreciated by others or not. This is a simple system used to find the user's opinion about the posted topic. This kind of system also work as an advertisement. It is a most efficient one and highly useful in the society.

7.2 Future Enhancements

The opinion mining system is a model that can be highly used by the users in the society in future. As it is a fast operation and efficient one, it can be mostly used by all the users who will be active in the social media. It can be converted into application from a simple model with some additional features.

Chapter 8

PLAGIARISM REPORT

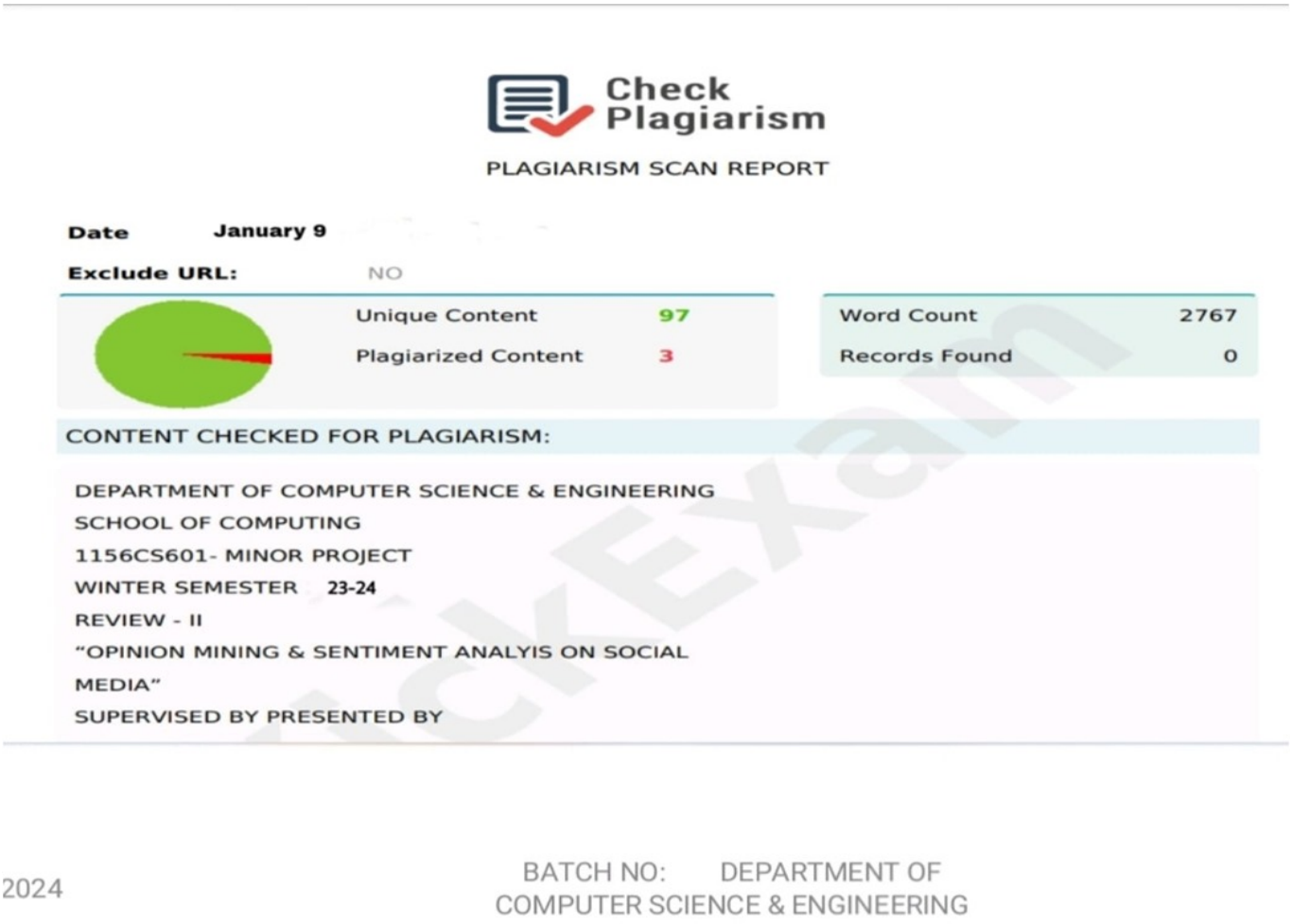


Figure 8.1: **Plagiarism Image**

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

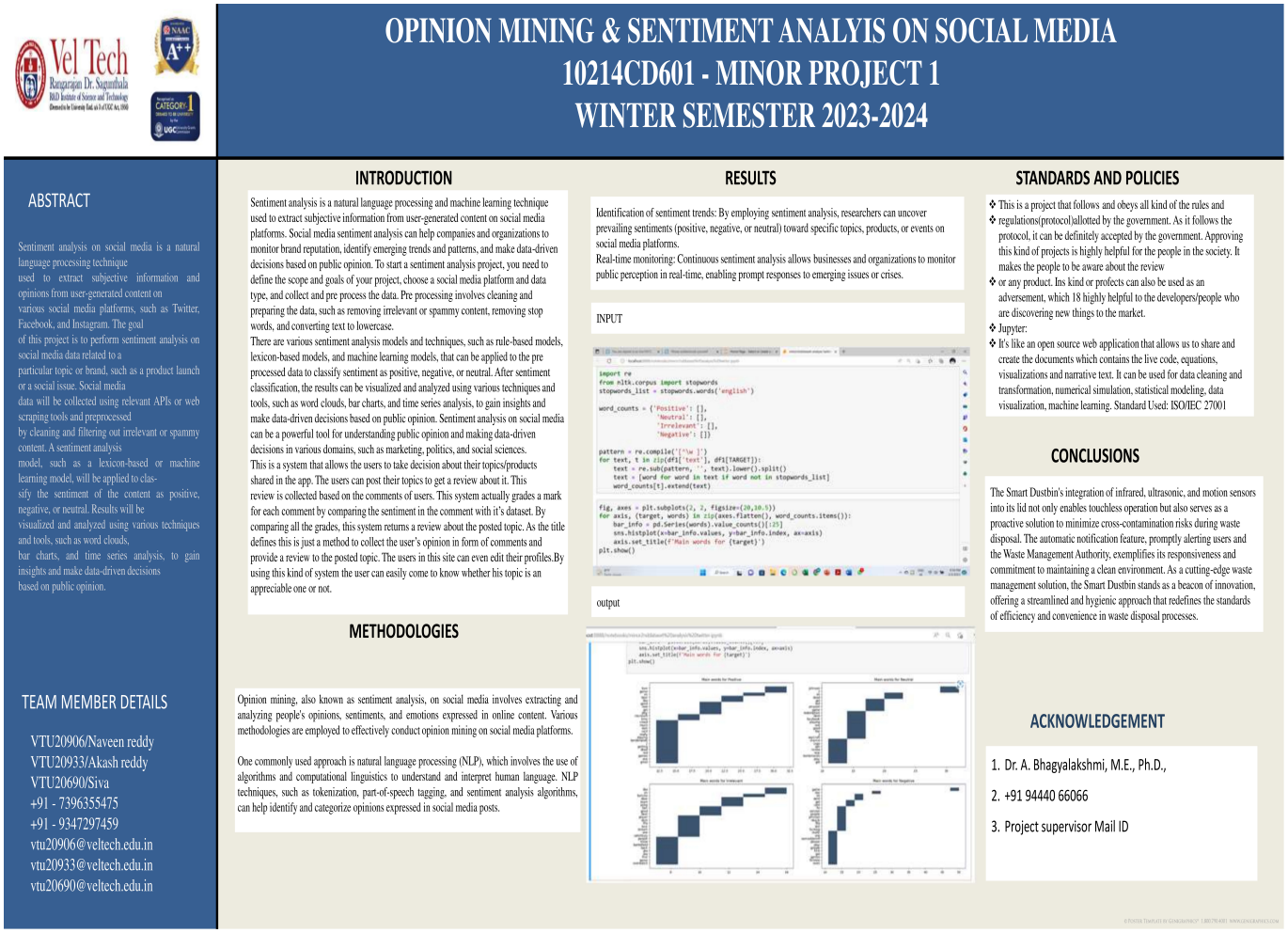


Figure 9.1: **Poster Presentation**

References

[1] Thai Kim Phung, Nguyen An Te, Tran Thi Thu Ha, "A machine learning approach for opinion mining online customer reviews", ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter), PP.243-246,2021.

[2] Rawan Fahad Alhujaili ,Wael M.S.Yafooz, "Sentiment Analysis for Youtube Videos with User Comments : Review", International Conference on Artificial Intelligence and Smart Systems (ICAIS), pp.814-820, 2021.

[3] Nourin Islam, Ms.Nasrin Akter,Abdus Sattar "Sentiment Analysis on Food Review using Machine Learning Approach", pp.157-164,2021.

[4] Nikhil Yadav, Omkar Kudale Srishti Gupta, "Twitter Sentiment Analysis Using Machine Learning For Product Evaluation ", International Conference on Inventive Computation Technologies (ICICT), pp.181-185,2020.

[5] Ritwik Murali, Akash Ravi, Harshit Agarwal, "Enhancing Digital Well-being using Opinion Mining and Sentiment Classifiers" International Conference on Inventive Computation Technologies (ICICT), pp.32-37, 2020.

[6] Jieyu Ding Featherstone, George A, "Validating Sentiment Analysis on Opinion Mining Using Self-reported Attitude Scores", Seventh International Conference on Social Networks Analysis, Management and Security (SNAMS), 2020.

[7] Ms. Binju Saju ,Ms.Siji Jose, MrAnmalAntorny "Comprehensive Study on Sentiment Analysis: Types, Approaches, Recent Applications, Tools and APIs", 2020.

[8] Sheresh Zahoor and Rajesh Rohilla, "Twitter Sentiment Analysis using Machine Learning Algorithms: A Case Study",IEEE, pp.194-199, 2020.

[9] H. Soong, N. B. A. Jalil, R. Kumar Ayyasamy and R. Akbar, “The Essential of Sentiment Analysis and Opinion Mining in SocialMedia : Introduction and Survey of the Recent Approaches and Techniques”, IEEE 9th Symposium on Computer Applications Industrial Electronics (ISCAIE), pp. 272-277, 2019.

[10] Kumar Ravi, Vadlamani Ravi, “A survey on opinion mining and sentiment analysis”, Knowledge based systems 89, pp. 14-46, 2019.

[11] Arti, Kamanksha Prasad Dubey,Sanjy Agarwal “An Opinion Mining For Indian Premier League Using Machine Learning Techniques”, IEEE, 2019.