A

PROJECT REPORT

ON

SENTIMENT ANALYSIS USING RANDOM FOREST MACHINE LEARNING ALGORITHM

SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE

OF

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

UNDER

SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

BY

AISHWARYA AHERGAWLI B190258501

ABHIJEET GIRASE B190258527

ISHWARI TALE B190258576

AYUSHI KHANBARAD B190258543

UNDER THE GUIDANCE

PROF. R. Y. TOTARE

DEPARTMENT OF INFORMATION TECHNOLOGY

ALL INDIA SHRI SHIVAJI MEMORIAL SOCIETY’S

INSTITUTE OF INFORMATION TECHNOLOGY

PUNE– 411001

APRIL 2023

AISSMS
INSTITUTE OF INFORMATION TECHNOLOGY
(IOIT)
ADDING VALUE TO ENGINEERING
An Autonomous Institute Affiliated to Savitribai Phule Pune University
Approved by AICTE, New Delhi and Recognised by Govt. of Maharashtra
Accredited by NAAC with "A+" Grade | NBA - 5 UG Programmes

सत्याला मरण नाही

SINCE 1917

Department of Information Technology

Certificate

This is to certify that the Project Report entitled

Sentiment Analysis Using Random Forest Machine Learning Algorithm

(Sustainable Development Goals Mapping- Industry, Innovation, and Infrastructure)

Submitted by

Aishwarya AherGawli, Abhijeet Girase, Ishwari Tale, Ayushi Khanbarad

is a record of bona-fide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering in Information Technology at All India Shri Shivaji Memorial Societies' Institute of Information Technology, Pune under the Savitribai Phule Pune University, Pune. This work is completed during the academic year 2022-23, under our guidance.

Prof. R. Y. Totare	Dr. Meenakshi A. Thalor	Dr. P.B.Mane
Project Guide	Head of Department	Principal

Examiner 1: - - - - - - -

Examiner 2: - - - - - - -

Contents

Acknowledgement

There is always a sense of gratitude that people express towards others for their help and supervision in achieving the goals. This formal piece of acknowledgment is an attempt to express the feeling of gratitude towards people who helped us in completing our project.

We would like to express our deep and sincere gratitude to our project guide, **Prof. R. Y. Totare** for allowing us to do this work and providing invaluable guidance. We would like to express our deep gratitude to **Dr. P. B. Mane, Principal**, **Dr. Meenakshi A. Thalor, Head of Department**, and **Prof. R. Y. Totare, Project Coordinator** for their constant co- operation. They were always there with competent guidance and valuable suggestions throughout the pursuance of this presentation.

We would also like to appreciate all the respondents and group members whose responses and coordination were of utmost importance for the presentation and who helped us a lot in collecting necessary information. Above all, no words can express our feelings to our parents, friends, and all those people who supported us during our project.

Aishwarya AherGawli
Abhijeet Girase
Ishwari Tale
Ayushi Khanbarad

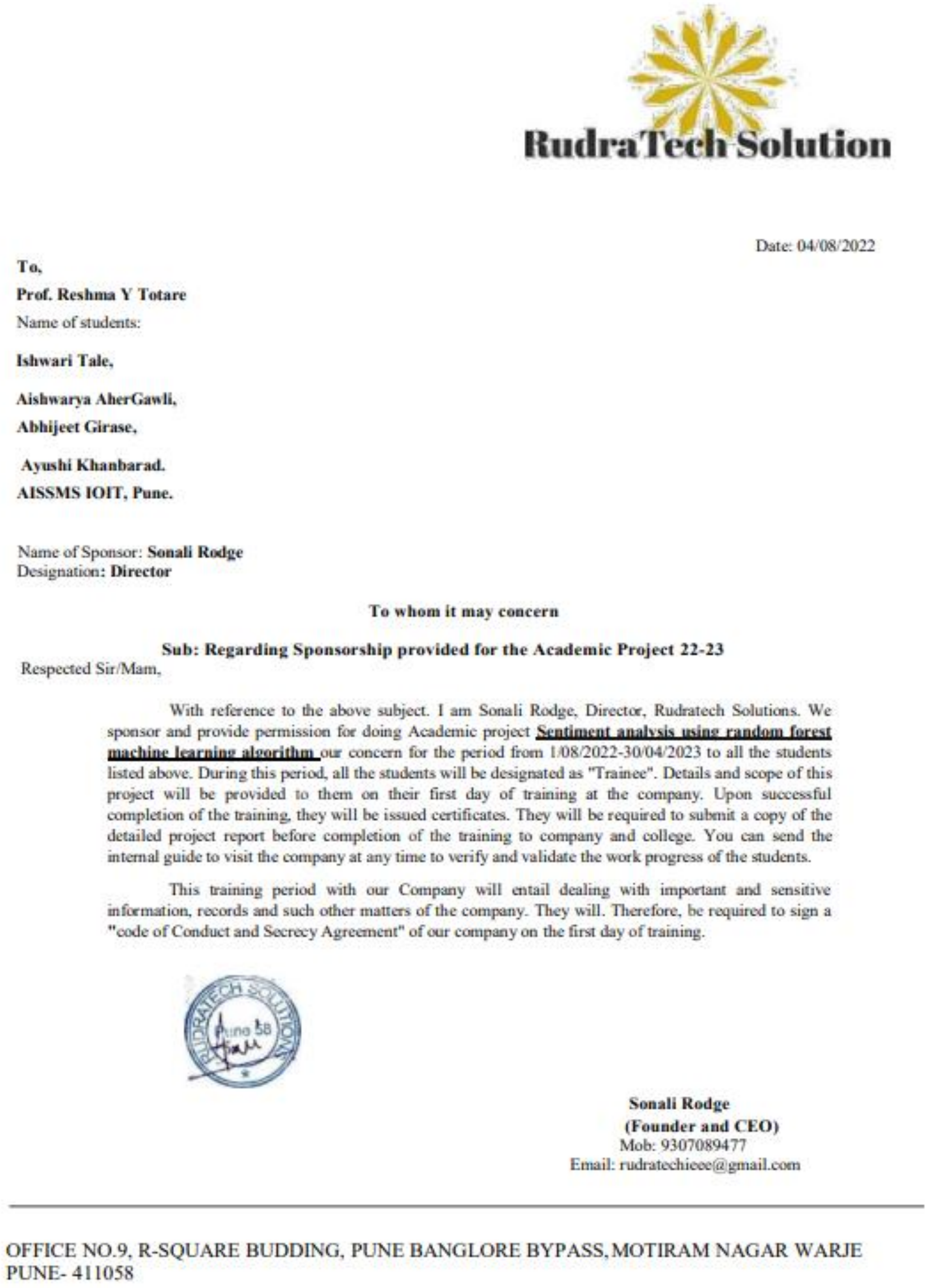
Abstract

Social Media sites like twitter have billions of people share their opinions day by day as tweets. As tweet is characteristic short and basic way of human emotions. So, in this paper we focused on sentiment analysis of Twitter data. Most of Twitter's existing sentiment analysis solutions basically consider only the textual information of Twitter messages and strives to work well in the face of short and ambiguous Twitter messages. Recent studies show that patterns of spreading feelings on Twitter have close relationships with the polarities of Twitter messages. In this paper we focus on how to combine the textual information of Twitter messages and sentiment dissemination models to get a better performance of sentiment analysis in Twitter data. To this end, proposed system first analyzes the diffusion of feelings by studying a phenomenon called inversion of feelings and find some interesting properties of the reversal of feelings. Therefore, we consider the interrelations between the textual information of Twitter messages and the patterns of diffusion of feelings, and we propose an iterative algorithm called SentiDiff to predict the polarities of the feelings expressed in Twitter messages. As far as we know, this work is the first to use sentiment dissemination models to improve Twitter's sentiment analysis. Numerous experiments in the real-world dataset show that, compared to state-of-the-art text- based analysis algorithms.

Keywords

Text Mining, Machine Learning, Sentiment Analysis, Sentiment Diffusion, Twitter

Sponsorship Certificate



RudraTech Solution

Date: 04/08/2022

To,

Prof. Reshma Y Totare

Name of students:

Ishwari Tale,

Aishwarya AherGawli,

Abhijeet Girase,

Ayushi Khanbarad.

AISSMS IOIT, Pune.

Name of Sponsor: **Sonali Rodge**
Designation**: Director**

To whom it may concern

Sub: Regarding Sponsorship provided for the Academic Project 22-23

Respected Sir/Mam,

With reference to the above subject. I am Sonali Rodge, Director, Rudratech Solutions. We sponsor and provide permission for doing Academic project **Sentiment analysis using random forest machine learning algorithm** our concern for the period from 1/08/2022-30/04/2023 to all the students listed above. During this period, all the students will be designated as "Trainee". Details and scope of this project will be provided to them on their first day of training at the company. Upon successful completion of the training, they will be issued certificates. They will be required to submit a copy of the detailed project report before completion of the training to company and college. You can send the internal guide to visit the company at any time to verify and validate the work progress of the students.

This training period with our Company will entail dealing with important and sensitive information, records and such other matters of the company. They will. Therefore, be required to sign a "code of Conduct and Secrecy Agreement" of our company on the first day of training.

RUDRATECH SOLUTIONS
Pune 58

Sonali Rodge
(Founder and CEO)
Mob: 9307089477
Email: rudratechieee@gmail.com

OFFICE NO.9, R-SQUARE BUDDING, PUNE BANGLORE BYPASS, MOTIRAM NAGAR WARJE
PUNE- 411058

List of Figures

List of Tables

Notation and Abbreviations

ML: Machine Learning

NLP: Natural Language Processing

SA: Sentiment Analysis

NB: Naïve Bayes

LR: Logistic Regression

DT: Decision Tree

SVM: Support Vector Machines

CNN: Convolutional Neural Network

LSTM: Long Short-Term Memory

MCNN-MA: Multi-Head Attention Mechanism

TF-IDF: Term Frequency-Inverse Document Frequency

Chapter 1

INTRODUCTION

1.1 Introduction

Twitter, a popular micro blogging service around the world, has shaped and transformed the way people get information from the people or organizations that interest them. On Twitter, users can post status update messages, called tweets, to tell their followers what they are thinking, what they are doing or what is happening around them. In addition, users can interact with another user by replying or republishing their tweets. Since its creation in 2008, Twitter has become one of the largest online social media platforms in the world. Given the increasing amount of data available from Twitter, the polarity of the feelings of mining users expressed in Twitter messages has become a hot research topic due to its wide applications. For example, in analyzing the polarities of Twitter users on political parties and candidates, different tools have been developed to provide strategies for political elections. Commercial companies also use Twitter sentiment analysis as a quick and effective way to monitor people's feelings about their products and brands.

This analysis is done by looking for opinions or sentiments from several sentences or tweets obtained. Therefore, this stack of text data in Twitter is quite valuable because it stores valuable information. To uncover this information, data mining needs to be done using certain techniques. Mining this data can be done using text mining techniques which can be combined also using the Natural Language Preprocessing approach. Furthermore, important data that has been mined needs to be determined by the type of sentiment. This is done by using analytical sentiments. Twitter is one type of social media that is often used. Users use Twitter to convey their Twitter to the general public. The number of Twitter users has reached 330 million people worldwide and every second produces 18000 data. The chirp delivered can be in the form of news, opinions, arguments, and several other types of sentences. This causes twitter to be rich in text that has certain data. In general, someone wants opinions from other people as input to determine decisions. This opinion can be done by asking directly. By asking directly, it takes time and effort to meet people who are believed to ask. Another way is to get opinions from Twitter. Opinions in the form of tweets provided by Twitter with a large amount. However, this opinion must be distinguished based on the type of positive, negative, and neutral opinions. In addition, these tweets have not been grouped according to the categories you want to find. So, it is still widespread and necessary.

In today's digital age, social media platforms like Twitter have become powerful tools for expressing opinions and sentiments on various topics. The vast amount of user-generated content on Twitter provides a valuable source of data that can be leveraged to understand public sentiment and analyze trends. Sentiment analysis, also known as opinion mining, aims to extract and classify sentiments expressed in text data.

The field of sentiment analysis, also known as opinion mining, has gained significant attention in recent years due to the explosion of social media and online communication platforms. Sentiment analysis aims to automatically identify and classify the sentiment expressed in a piece of text, such as positive, negative, or neutral. This information is valuable for businesses, governments, and individuals alike, as it provides insights into public opinion, customer satisfaction, and brand perception.

1.2 Motivation

- In today's digital era, people express their opinions and emotions through various online platforms, including social media, product reviews, and discussion forums.
- The ability to automatically analyze and understand these sentiments has numerous applications. Businesses can monitor customer feedback and sentiment towards their products or services, enabling them to make informed decisions and improve their offerings.
- Governments can gauge public sentiment towards policies and initiatives, helping them assess public sentiment and address concerns. Individuals can use sentiment analysis to analyze and understand public opinion, support decision-making, or track their own sentiments over time.

- Text sentiment analysis is the process of analyzing, processing, summarizing, and judging the sentiment tendency of subjective texts with sentimental colors.
- It aims at mining 'the sentiment polarity in text information and has become a hot issue in the field of natural language processing (NLP) in recent years.
- Twitter sentiment analysis has gained significant attention due to its potential applications in various domains. It has proven useful in political analysis, customer feedback analysis, market research, and brand management.
- By applying machine learning algorithms, such as Random Forest, to sentiment analysis, we can automate the process and handle large volumes of data efficiently.

1.3 Related Theory

Sentiment analysis (SA) is one of the most studied research areas combining natural language processing, data mining, and web mining. Owing to its importance to business and community, SA research has spread into management and social sciences as discussed by Liu. At present, there are three kinds of research methods for text sentiment analysis: methods based on sentiment dictionaries, methods based on traditional machine learning, and methods based on deep learning. The method based on the sentiment dictionary assigns a polarity score to each word in the dictionary after the sentiment dictionary is established, and then matches the words in the sentence with the dictionary to obtain the corresponding polarity score, and finally aggregates the polarity scores of all words (such as averaging) to get the final sentiment polarity of the text.

Various ML algorithms have been used to conduct SA in the restaurant's domain. NB, LR, and DT. ML algorithms were applied by Hassan et al.to conduct SA on three different datasets, namely, the Yelp dataset, IMDB dataset, and Arabic qaym.com restaurant reviews dataset. Performance was measured in terms of accuracy and recall. NB and LR recorded the best results. Similarly, NB, SVM, multilayer perceptron, DT, k-NN, and fuzzy logic were applied by Kumar and Jaiswal on data extracted from Twitter and Tumblr, which are widely used micro-blogging social networks. A comparative analysis of performance is presented in terms of precision, recall, and accuracy. In today's digital age, social media platforms like Twitter have become powerful tools for expressing opinions and sentiments on various topics. The vast amount of user-generated content on Twitter provides a valuable source of data that can be leveraged to understand public sentiment and analyze trends. Sentiment analysis, also known as opinion mining, aims to extract and classify sentiments expressed in text data.

1.4 Need of the topic

Sentiment analysis can improve customer loyalty and retention through better service outcomes and customer experience. The importance of customer sentiment extends to what positive or negative sentiment the customer expresses, not just directly to the organization, but to other customers as well. People commonly share their feelings about a brand's products or services, whether they are positive or negative, on social media. If a customer likes or dislikes a product or service that a brand offers, they may post a comment about it -- and those comments can add up. Such posts amount to a snapshot of customer experience that is, in many ways, more accurate than what a customer survey can obtain.

This is where social media monitoring comes in. By mining the comments that customers post about the brand, the sentiment analytics tool can surface social media sentiments for natural language processing, yielding insights. This activity can result in more focused, empathetic responses to customers.

Sentiment analysis is a valuable technique used in natural language processing to determine the sentiment or opinion expressed in a piece of text. It has a wide range of applications, including social media monitoring, customer feedback analysis, market research, and brand reputation management. Random Forest algorithm is one of the popular machine learning algorithms used for sentiment analysis. Here are some reasons why Random Forest is a suitable choice for sentiment analysis:

i. Feature Importance- Random Forest provides a measure of feature importance, which can clarify which features contribute the most to predicting sentiment. This information can guide feature selection and help in identifying the key indicators of sentiment in the text data.

ii. Scalability-: Random Forest can handle large datasets with high-dimensional feature spaces efficiently. With the increasing volume of textual data available, scalability is an important consideration for sentiment analysis algorithms.

iii. Interpretability- Although Random Forest is an ensemble of decision trees, it still provides a level of interpretability. You can analyze individual decision trees within the forest to understand the reasoning behind the model's predictions. This can be useful for explaining the sentiment analysis results to stakeholders.

iv. Robustness to Noise: Random Forest is known for its robustness to noisy data and outliers. In sentiment analysis, the text data can often be noisy due to typographical errors, abbreviations, slang, or misspellings. Random Forest can handle such noisy inputs and still produce reliable predictions.

v. Ensemble Learning: Random Forest is an ensemble learning algorithm that combines multiple decision trees to make predictions. Each decision tree is trained on a different subset of the data, and their predictions are aggregated to obtain the result. This ensemble approach helps to reduce overfitting and improves the generalization capability of the model.

vi. Handling Nonlinear Relationships: Sentiment analysis often involves capturing complex and nonlinear relationships between the text features and the sentiment labels. Random Forest can effectively handle such relationships by partitioning the feature space into different regions, allowing for more flexible modeling compared to linear models.

1.5 Organization Of the Chapters

Chapter 1: Introduction

This chapter introduces sentiment analysis using the Random Forest algorithm for Twitter data. It discusses the significance of sentiment analysis, its applications, and the motivation behind the research.

Chapter 2: Literature Survey

This chapter presents an analysis of relevant research papers on sentiment analysis, highlighting their findings, limitations, and key learnings.

Chapter 3: Proposed Work

This chapter defines the problem and scope of the project, outlining the objectives and constraints. It focuses on building a sentiment analysis system using the Random Forest algorithm.

Chapter 4: Research Methodology

This chapter explains the research methodology employed, including data collection, preprocessing, feature extraction, sentiment diffusion analysis, and the application of the Random Forest algorithm.

Chapter 5: Project Design

This chapter covers the hardware and software requirements, risk analysis, data flow diagrams, project schedules, and UML diagrams related to the sentiment analysis system.

Chapter 6: System Implementations

This chapter discusses the implementation details of the sentiment analysis system, including the libraries, functions, and algorithms used.

Chapter 7: System Testing

This chapter introduces software testing and emphasizes the creation of test cases to ensure the proper functioning of the sentiment analysis system.

Chapter 8: Experimental Results

This chapter presents the experimental results of sentiment analysis using the Naïve Bayes and Random Forest algorithms, comparing the proposed system's performance with the existing method.

Chapter 9: Conclusion

This chapter concludes the project report, summarizing the findings, methodology, and contributions. It also suggests future research directions and enhancements.

Chapter 10: Future Scope

This chapter explores potential areas for further development and research in sentiment analysis, including live tweet classification, emoji predictions, and identifying ironic statements.

Chapter 11: References

Chapter 11 of the report provides a list of references used throughout the text for further reading and research. The chapter includes a collection of academic papers, articles, and conference proceedings that have contributed to the knowledge and development of sentiment analysis.

Chapter 2

LITERATURE SURVEY

To Understand the Problem Statement in Depth We Referred to The Below-Given Reference Papers in Table 1 and Mentioned the Findings, Learnings, And Drawbacks.

Table 1: Literature Survey

Sr. No.	**Paper Details**	**Methodology**	**Gap**	**Conclusion**
1.	Short Text Sentiment Analysis Based on Multi- Channel CNN With Multi-Head Attention Mechanism	Multi-channel convolutional neural network	. CNNs tend tobe much slower because of operations like maxpool. In case the convolutional neural network is made up of multiple layers, the training process could take a particularly long time if the computer does not have a good GPU.	This paper proposes a sentiment analysis model MCNN-MA that combines multi-channel convolutional neural network and multi-head attention mechanism. This model builds sentiment features, and combines sentiment features to form a three-channel input. Finally, the sentiment classification result is obtained through the sentiment classification layer. The model in this paper is based on a multi-channel convolutional neural network, which has obvious advantages in training time compared to the model based on LSTM network

2.	A Study of the Application of Weight Distributing Method Combining Sentiment Dictionary and TF- IDF for Text Sentiment Analysis	A sentence vector generation method based on sentiment dictionaries and pre-trained word vectors	The counter vector has inability to identifying more important and less important words for analysis. It just considers words that are abundant in a corpus as the most statistically significant word.	In this paper, a sentence vector generation method based on sentiment dictionaries and pre- trained word vectors is proposed for sentiment classification, which calculates the weights of sentiment words and neutral words in a sentence separately, and retains the overall information of the sentence while highlighting the sentiment words. From the experimental result, the accuracy of text sentiment analysis using the method proposed in this paper reaches 2.1%, which is 13.9% and 7.7% higher respectively than the other two methods.
3.	Sentiment Analysis of Customer Reviews Using a Hybrid Evolutionary SVM-Based Approach	Tokenization methods, including 1-Gram, 2-Gram, 3-Gram, and bag-of-word; A hybrid optimization technique comprising PSO and SVM	One of the biggest challenges in the tokenization is the getting the boundary of thewords. In English the boundary of theword is usually defined by a space and punctuation marks define the boundary of the sentences.	We produced four versions of the collected dataset using different tokenization methods, including 1-Gram, 2-Gram, 3-Gram, and bag-of-words. Further, we implemented a hybrid optimization technique comprising PSO and SVM to findthe best weights while also finding the k values of four different oversampling techniques to predictthe sentiments of reviews. The studydemonstrates that the proposed PSO-SVM approach is effective and outperforms the other

	in an Imbalanced Data Distribution.		In languages such as Chinese, Korean, Japanese symbols represent thewords and it is difficult to get the boundary of the words.	approaches in all investigated measures (accuracy, F-measure, g-means, and AUC). In more detail, the PSO-SVM providedbetter results than the standard SVM, LR, RF, DT, k-NN, and XGBoost in all versions of thedatasets.
4.	RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis with Transformer and Re current Neural Network	Transformer and Recurrent Neural Network	Computation is slow. Faces issues like exploding or gradient vanishing.	This paper presents a hybrid model of Transformer and Recurrent Neural Network, referred to as the RoBERTa-LSTM model. Firstly,data augmentation with GloVe pre- trained word embedding is used to generate more lexically similar samples and to oversample the minority classes. The experimental results demonstrate that the proposed RoBERTa-LSTM model outshines the state-of-the-art methods in sentiment analysis on IMDb dataset, Twitter US Airline Sentiment dataset, and Sentiment140 dataset.

Chapter 3

PROPOSED WORK

3.1 Problem Definition

To build and develop a sentiment analysis based on short textual information and diffusion patterns using random forest machine learning algorithms.

3.2 Project Scope

- The ability to detect and track a user's state of mind has the potential to allow a computingsystem to offer relevant information when a user needs help not just when the user requests help.
- Find sentiment polarities expressed in Twitter messages.
- Find sentiments i.e., positive, negative, and neutral.

3.3 Project Objectives

The main aim of this project report are as follows**:**

- To collect a relevant dataset of tweets from Twitter's API.
- To preprocess the collected data to remove noise, handle special characters, normalize the text.
- To perform feature extraction to represent tweets as numerical vectors suitable for machine learning algorithms.
- To train a Random Forest classifier on the labeled data to predict sentiment labels.
- To evaluate the performance of the model using appropriate metrics.
- To analyze and interpret the results to gain insights into the sentiment patterns.
- To classify the sentiment polarity of a Twitter message as positive, neutral, or negative.
- To test system using real streaming using Twitter API.
- Try to improve accuracy using machine learning algorithm.
- To help people in sentiments-related research to improve the processing of data.

3.4 Project Constraints

While Twitter sentiment analysis offers valuable insights, there are several constraints and challenges that need to be considered:

- Limited Context: Twitter posts, or tweets, have a maximum character limit of 280 characters. This limited space often results in short and fragmented text, making it challenging to capture the full context and nuances of sentiment. The brevity of tweets can lead to ambiguity and difficulties in accurately interpreting sentiment.
- Noisy and Informal Language: Twitter is known for its informal language, abbreviations, slang, misspellings, and emoticons. This can introduce noise and ambiguity into the text, making sentiment analysis more challenging. Understanding and accurately interpreting such informal language poses difficulties for sentiment analysis algorithms.
- Contextual Ambiguity: Tweets often lack explicit context. The same tweet can have different sentiment interpretations depending on the context or the topic being discussed. Without additional context, it can be challenging to accurately determine the sentiment behind a tweet.
- Irony and Sarcasm: Twitter users frequently use irony, sarcasm, and other forms of figurative language to express sentiment. Identifying and interpreting these nuanced forms of expression accurately is a complex task for sentiment analysis algorithms.
- Data Preprocessing: Twitter data requires specific preprocessing steps to handle hashtags, mentions, URLs, and emoticons. Removing noise, normalizing text, and handling these Twitter-specific elements require careful preprocessing techniques to improve the accuracy of sentiment analysis.
- Data Volume and Velocity: Twitter generates a vast amount of data in real-time, with millions of tweets posted every day. Analyzing such a large volume of data and processing it in real-time can be computationally intensive and require scalable infrastructure.
- Biased Data: Twitter data can be subject to biases, including sampling biases, selection biases, or biases based on the demographic or user preferences of the platform. Biased data can lead to skewed sentiment analysis results, affecting the accuracy and reliability of the analysis.

- Privacy and Ethical Concerns: Twitter sentiment analysis involves analyzing publicly available data, but privacy concerns may arise when dealing with user-generated content. It is essential to adhere to ethical guidelines and data protection regulations when collecting and analyzing Twitter data.
- Subjectivity and Opinion Variability: Sentiment analysis is inherently subjective, as different individuals may interpret the same tweet differently. Moreover, sentiment can vary among different user groups, cultures, or regions. Building a sentiment analysis model that captures this variability is a challenge.

Addressing these constraints requires careful consideration of the data preprocessing techniques, choice of sentiment analysis algorithms, and the integration of domain knowledge to enhance the accuracy and reliability of sentiment analysis on Twitter data.

Chapter 4

RESEARCH METHODOLOGY

4.1 System Architecture

The system architecture consists of the components as shown in the figure 4.1 such as Tweets extraction from twitter, preprocessing of data, feature extraction, Training set are defined for the given analysis. The training set is obtained by predefined set of positive or negative tweets which can be done using random forest and output obtained is positive, negative tweets. The Classifier will classify the tweets according the training set and regulates the polarity of the tweet as the output.

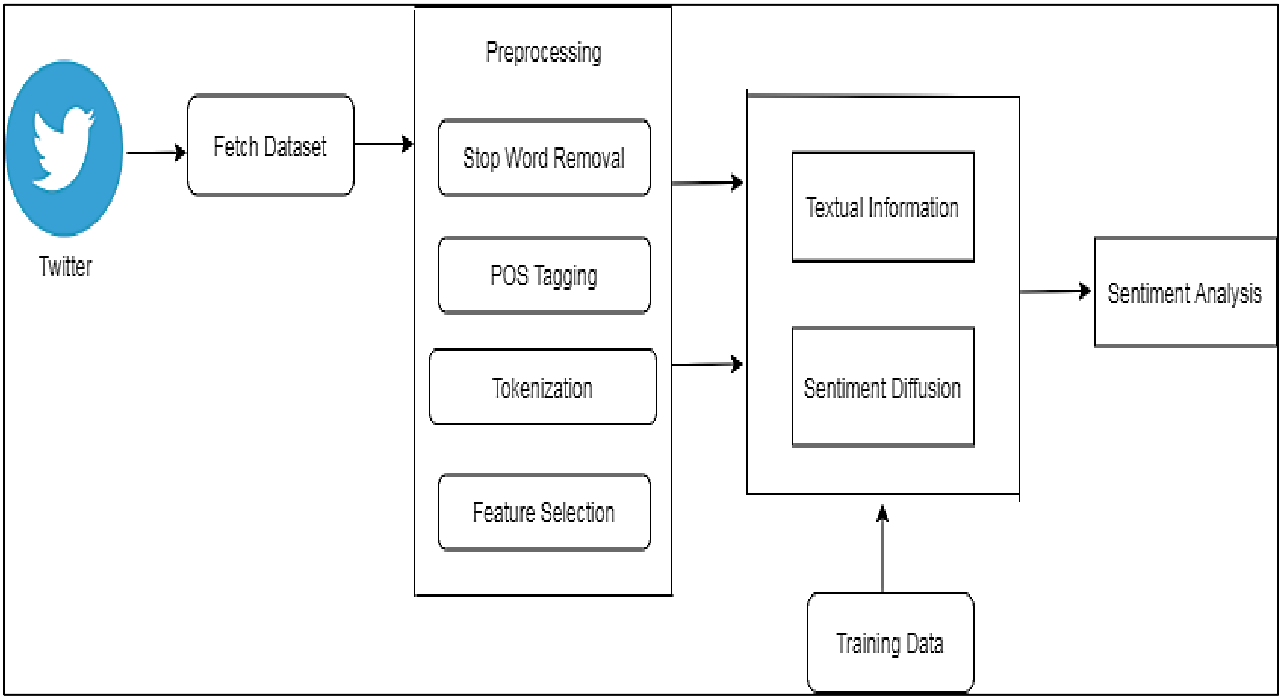


Fig. 4.1 System Architecture

4.1.1 Modules Split-up

- Data Collection:

  Data collection is the process of gathering, measuring, and analyzing accurate data from a variety of relevant sources to find answers to research problems, answer questions, evaluate outcomes, and forecast trends and probabilities. Accurate data collection is necessary to make informed business decisions, ensure quality assurance, and keep research integrity.

- Data Preprocessing:

  Data preprocessing, a component of data preparation, describes any type of processing performed on raw data to prepare it for another data processing procedure. It has traditionally been an important preliminary step for the data mining process.

- Feature Extraction:

  In machine learning, pattern recognition, and image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Feature extraction is related to dimensionality reduction.

- Classification:

  The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations based on training data**.** In Classification, a program learns from the given dataset or observations and then classifies new observation into several classes or groups.

4.1.2 Fetch dataset from twitter:

There are various sources where you can find publicly available datasets for sentiment analysis on Twitter. Here are a few options:

1. Sentiment140: The Sentiment140 dataset is a widely used dataset for sentiment analysis, containing 1.6 million labelled tweets.
2. Kaggle: Kaggle is a platform that hosts machine learning competitions and provides datasets for various tasks. You can search for sentiment analysis datasets related to Twitter on Kaggle.
3. Twitter API: If you have access to the Twitter API, you can fetch real-time tweets based on specific criteria or keywords.

4.1.3 Data Processing:

Data processing is an essential step in preparing your data for sentiment analysis or any other machine learning task. Here are some common data processing steps you can follow:

1. Data Cleaning:
   - Remove any irrelevant or unnecessary information from the dataset, such as metadata or columns that are not relevant to sentiment analysis.
   - Handle missing data: Decide how to handle missing values, whether by removing rows with missing values or imputing missing values with appropriate techniques.
   - Remove duplicates: Check for and remove any duplicate entries in the dataset.

2. Text Pre-processing:
   - Tokenization: Split the text into individual words or tokens.
   - Lowercasing: Convert all text to lowercase to ensure consistency.
   - Removing noise: Remove special characters, URLs, hashtags, and other noise that may not contribute to sentiment analysis.

- Stop word Removal: Eliminate common words (e.g., "the," "and," "is") that typically do not carry much sentiment information.
- POS tagging: Part-of-speech (POS) tagging is a natural language processing task that involves assigning grammatical labels or tags to words in each text. POS tags represent the syntactic category or function of a word within a sentence, such as noun, verb, adjective, etc. POS tagging is a crucial step in many languages processing applications, including sentiment analysis, text summarization, and machine translation.
- Tokenization: Tokenization is the process of splitting a text into individual words or tokens. In natural language processing (NLP), tokenization is an essential step as it serves as the foundation for various text analysis tasks. Tokenization allows the text to be broken down into meaningful units, such as words or sub words, which can then be further processed or analyzed.
- Lemmatization or Stemming: Reduce words to their base or root form to avoid redundancy and improve analysis accuracy.
- Encoding: Convert categorical variables, such as sentiment labels, into numerical representations for machine learning algorithms.

3. Feature Extraction:
   - Vectorization: Convert the pre-processed text into numerical feature vectors that machine learning algorithms can understand. Common techniques include Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), or word embeddings like Word2Vec or GloVe.
   - Feature Scaling: Normalize or scale the feature vectors to ensure all features have similar ranges, which can improve the performance of some machine learning algorithms.

4. Train-Test Split:
   - Split your processed dataset into training and testing sets. The training set is used to train your sentiment analysis model, while the testing set is used to evaluate its performance.

5. Additional Steps (if required):
   - Handling class imbalance: If your sentiment analysis dataset has an imbalance in the distribution of sentiment labels (e.g., many more positive tweets than negative ones), you may need to apply techniques such as oversampling, under sampling, or class weighting to address the imbalance.
   - Advanced pre-processing techniques: Depending on the specific characteristics of your data, you might consider other pre-processing techniques such as handling emojis, spell checking, or sentiment-specific transformations.

Remember that the specific data processing steps may vary depending on your dataset and the requirements of your sentiment analysis task. It is important to understand your data and perform exploratory data analysis to determine the appropriate processing steps for your particular use case.

4.1.4 Sentiment diffusion:

Sentiment diffusion refers to the process by which sentiments or opinions spread or propagate through a network or population. It involves the transmission and influence of sentiments from one individual or entity to others within a social network or community.

In the context of social media, sentiment diffusion can occur through various mechanisms:

1. Retweets and shares: When a user share a post or tweet containing a sentiment, it can reach a wider audience and potentially influence the opinions of others who come across it.

2. Comments and replies: Interactions such as comments, replies, or discussions on social media platforms allow sentiments to be exchanged and potentially influence the sentiment of other participants.

3. Network connections: Sentiments can spread through network connections or relationships. If a person with a particular sentiment is connected to others in the network, their sentiment may influence those connections.

4. Influencers: Influential individuals or opinion leaders with a large following can have a significant impact on sentiment diffusion. Their sentiments and opinions can spread rapidly among their followers and beyond.

5. Viral content: Particularly powerful or emotionally charged content has the potential to go viral, reaching many people quickly and potentially influencing their sentiments.

Understanding sentiment diffusion is important in social media analysis, marketing, and public opinion research. It helps to identify influential users, track the spread of sentiments, and analyse the dynamics of sentiment change within a network or population.

Researchers and data analysts use various techniques and models, such as network analysis, diffusion models, and sentiment analysis, to study sentiment diffusion and its impact on different social phenomena. By analysing the patterns and mechanisms of sentiment diffusion, researchers can gain insights into the dynamics of public sentiment, the formation of opinions, and the effects of social influence.

4.1.5 Sentiment analysis:

Sentiment analysis, also known as opinion mining is a natural language processing technique that aims to determine the sentiment or subjective opinion expressed in a piece of text. It involves analyzing text data to identify and classify the sentiment or emotional tone conveyed by the author. The objective of analysis is to understand underlying sentiment polarity, which is positive, negative, or neutral.

4.2 Methodology/algorithm Detail

The algorithm used here is Random Forest. Random Forest is the most popular andpowerful algorithm of machine learning.

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

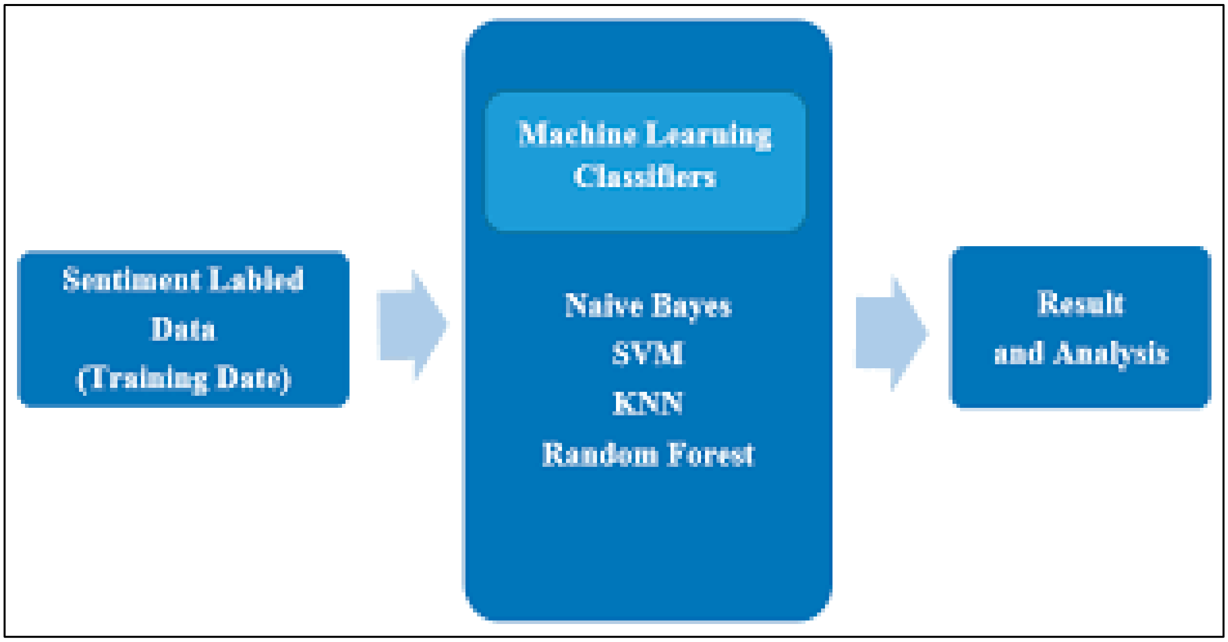

Sentiment Labled Data (Training Date)
Machine Learning Classifiers
Naive Bayes
SVM
KNN
Random Forest
Result and Analysis

Fig. 4.2 ML algorithm

4.2.1 Features of Random Forest:

- Miscellany: Each tree has a unique attribute, variety and features concerning other trees. Not all trees are the same.

- Immune to the curse of dimensionality: Since a tree is a conceptual idea, it requires no features to be considered. Hence, the feature space is reduced.

- Parallelization: We can fully use the CPU to build random forests since each tree is created autonomously from different data and features.

- Train-Test split: In a Random Forest, we do not have to differentiate the data for train and test because the decision tree never sees 30% of the data.

- Stability: The result is based on Bagging, meaning the result is based on majority voting or average.

4.2.2 Why use random forest?

There are a lot of benefits to using Random Forest Algorithm, but one of the main advantages is that it reduces the risk of overfitting and the required training time. Additionally, it offers a high level of accuracy. Random Forest algorithm runs efficiently in large databases and produces highly accurate predictions by estimating missing data.

Advantages of random forest are:

- Can perform both Regression and classification tasks.
- Produces good predictions that can be understood easily.
- Can handle large data sets efficiently.
- Provides a higher level of accuracy in predicting outcomes over the decision algorithm.

Chapter 5

PROJECT DESIGN

The overall design of the system is represented here which includes the hardware, software, risk for the system, flow of the data in the system and the documentation.

5.1 Hardware Requirements

1. Processor	- Intel i5
2. Speed	- 3.1 GHz
3. RAM	- 4 GB (min)
4. Hard Disk	- 40 GB
5. Key Board	- Standard Windows Keyboard
6. Mouse	- Two or Three Button Mouse
7. Monitor	- SVGA

5.2 Software Requirements

1. Operating System	-Windows 7/8/10
2. Front End	- HTML, JDK 1.8
3. Language	-Java
4. Server-side Script	- Java Server Pages
5. Database	-My SQL5.0
6. IDE	-Eclipse Oxygen, Visual Studio

5.3 Risk Analysis

- Overall traffic load
- Data loss
- Information security and security on technical like network, device, OS etc.
- Data damage

5.4 Data Flow Diagrams

DFD Level 0-

Data flow diagram level 0 show the flow of the data with respect to both the actions carried out the user as well as the admin.

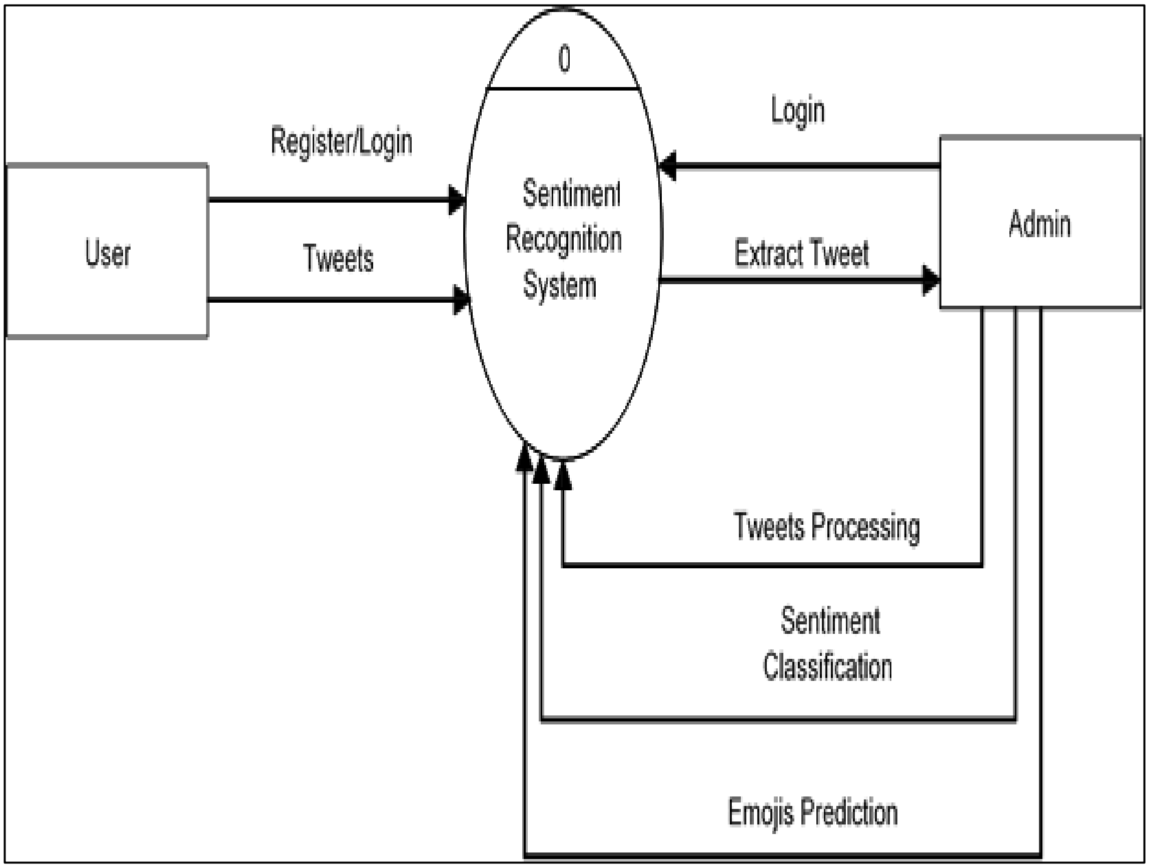


Fig. 5. 4. 1 DFD Level 0 Diagram

DFD Level 1-

Data flow Diagram level 1 shows the representation with respect to the activities taken placed by the user.

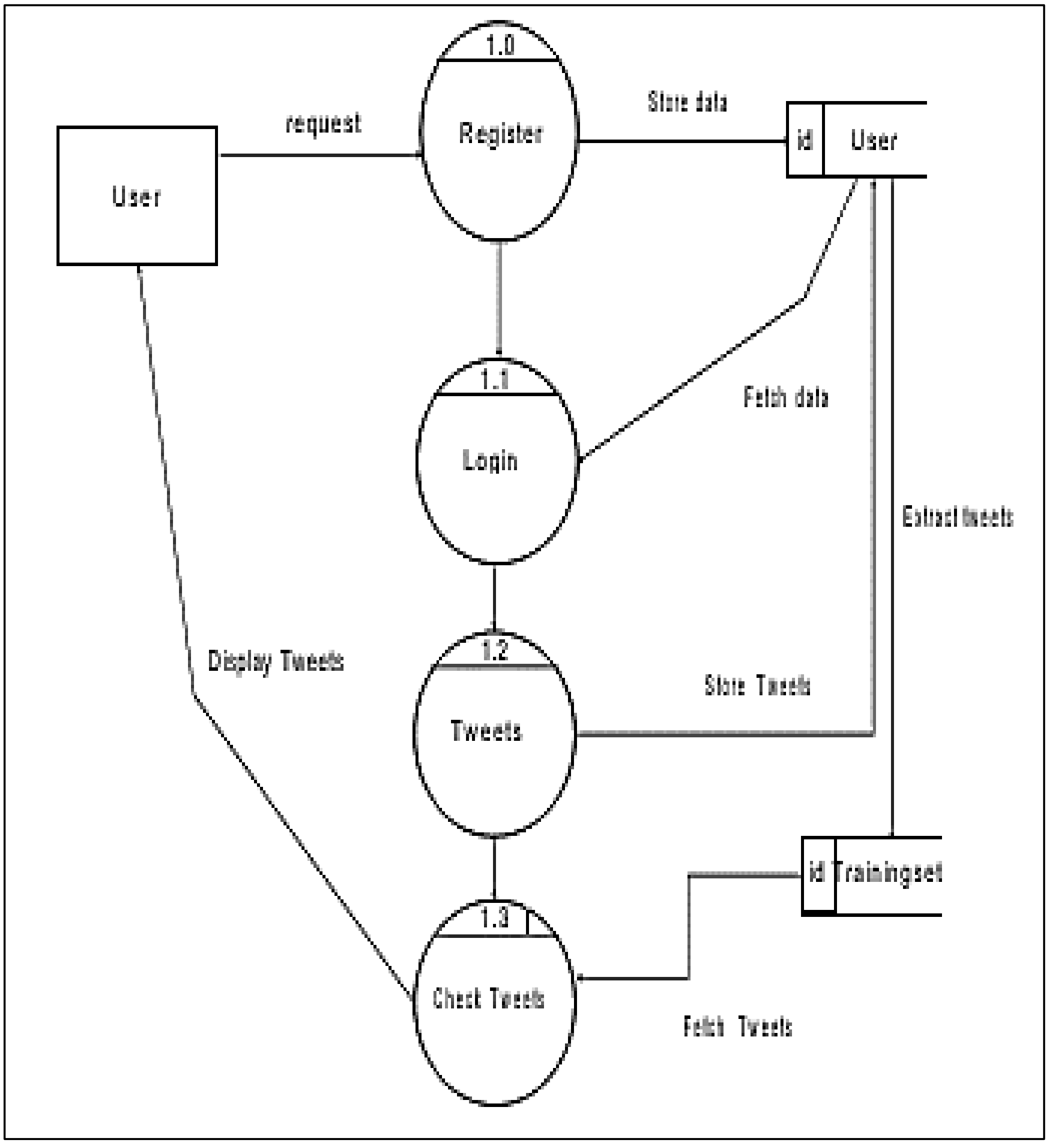


Fig. 5. 4. 2 DFD Level 1 Diagram

DFD Level 2-

In Data Flow Diagram level 2 the steps/activities carried out by the admin is represented. All the steps followed by the admin i.e., Login to the webpage, extract tweets, processing of tweets, prediction etc. has been shown.

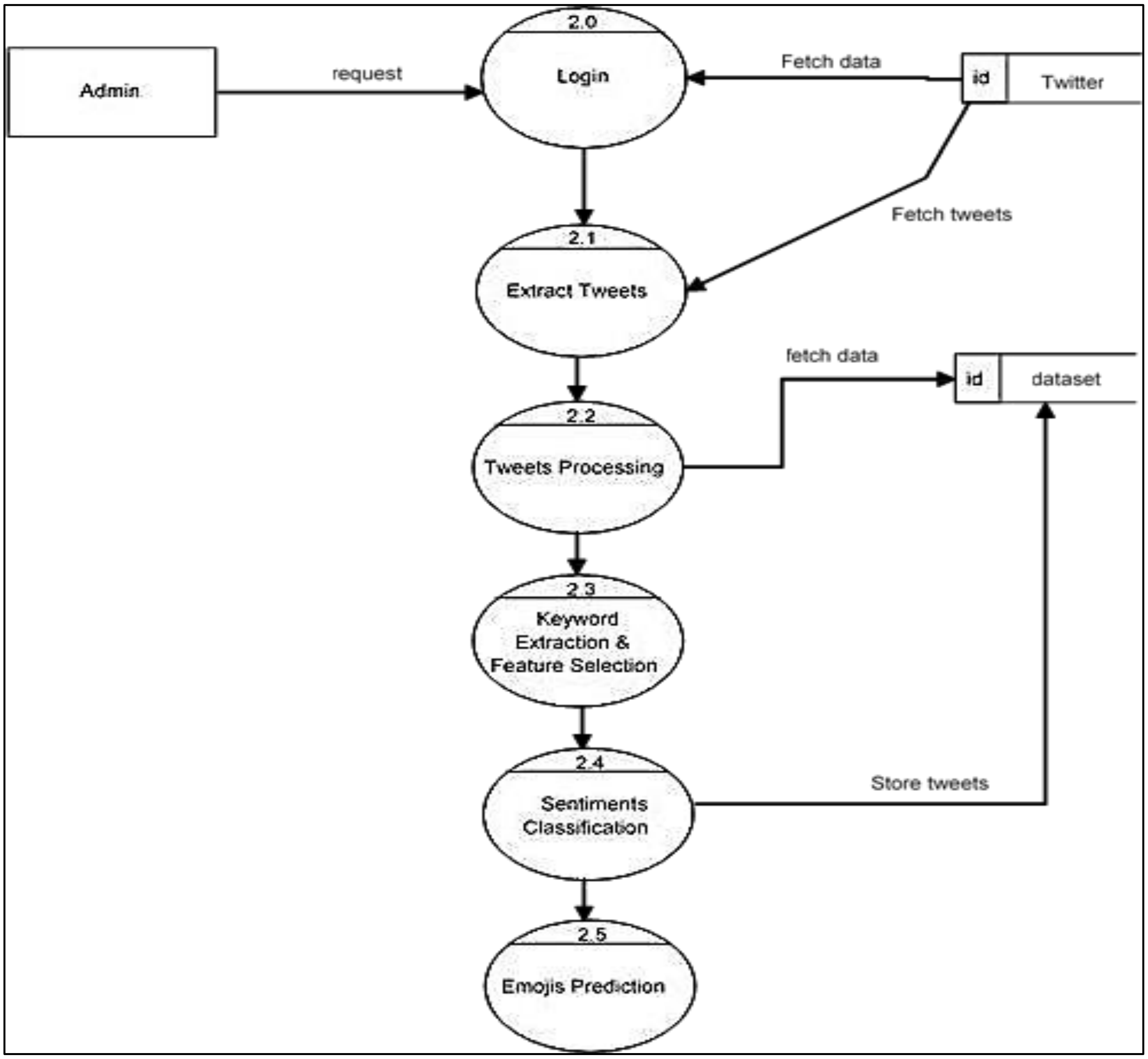


Fig. 5. 4. 3 DFD Level 2 Diagram

5.5 Project Schedules

The above table shows the overall activities of project carried out with respect to time duration. It contains activities from Topic finalization to the submission of Final Report.

Table 2: Project Schedule

Schedule		**Date**	**Project Activity**
July	1st Week	01/07/2022	Project Topic Searching
	2nd Week	08/07/2022	Project Topic Selection
	3rd Week	15/07/2022	Synopsis Submission
August	1st Week	05/08/2022	Presentation On Project Ideas
	2nd Week	12/08/2022	Submission Of Literature Survey
	3rd Week	19/08/2022	Review 1
September	1st Week	02/09/2022	Documentation for paper publishing.
	3rd Week	16/09/2022	Design Of Mathematical Model
	4th Week	23/09/2022	Paper is published
October	1st Week	09/10/2022	Report Preparation and Submission
November	1st Week	1/11/2022	Review 2
February	1st Week	6/2/2023	Review 3
March	3rd Week	20/3/2023	Review 4

5.6 UML Design and Documentation

Use Case Diagram:

Use case diagrams give a graphic overview of the actors involved in a system, different functions neededby those actors and how these different functions interact. It is a great starting point for any project discussion because you can easily identify the main actors involved and the main processes of the system.

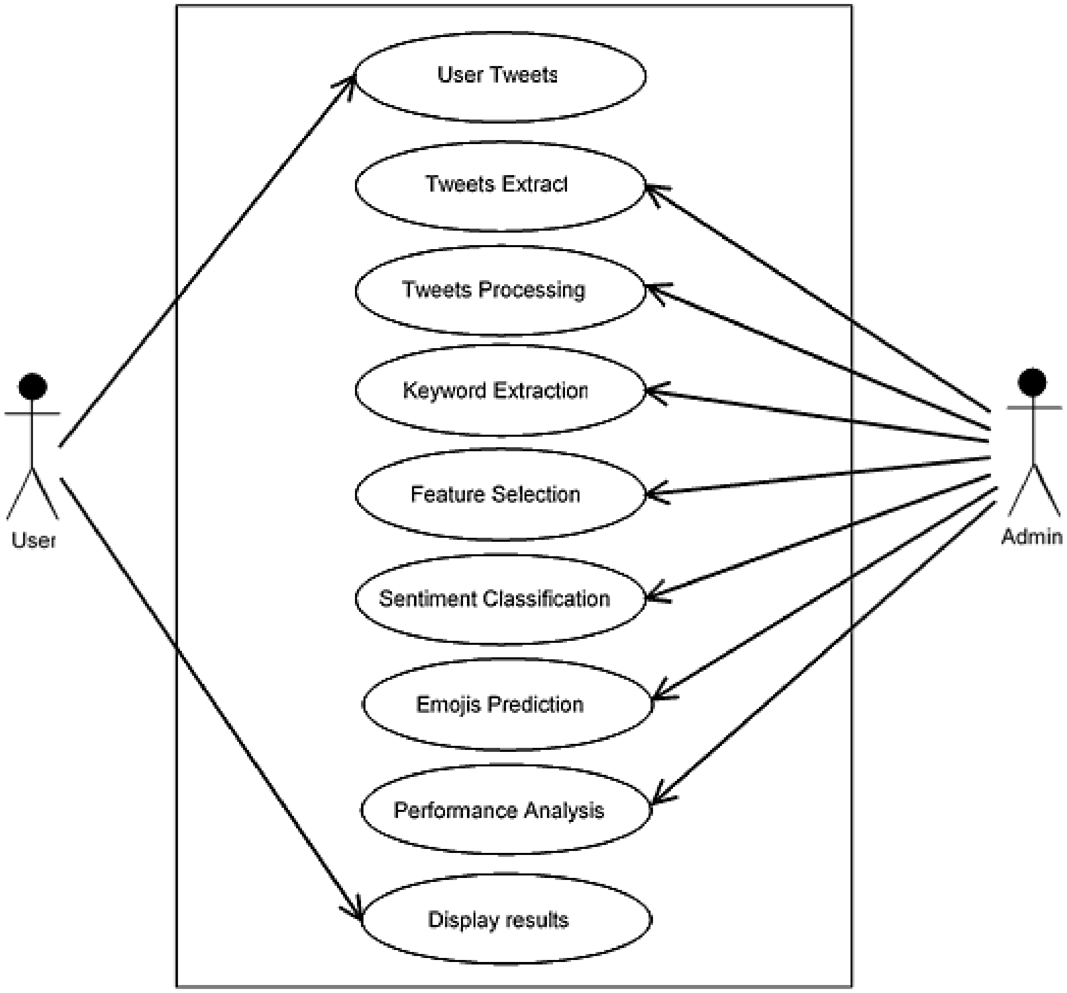


Fig. 5. 6. 1 Use Case Diagram

Sequence Diagram:

Sequence diagrams in UML show how objects interact with each other and the order those interactions occur. Sequence diagram for our site is:

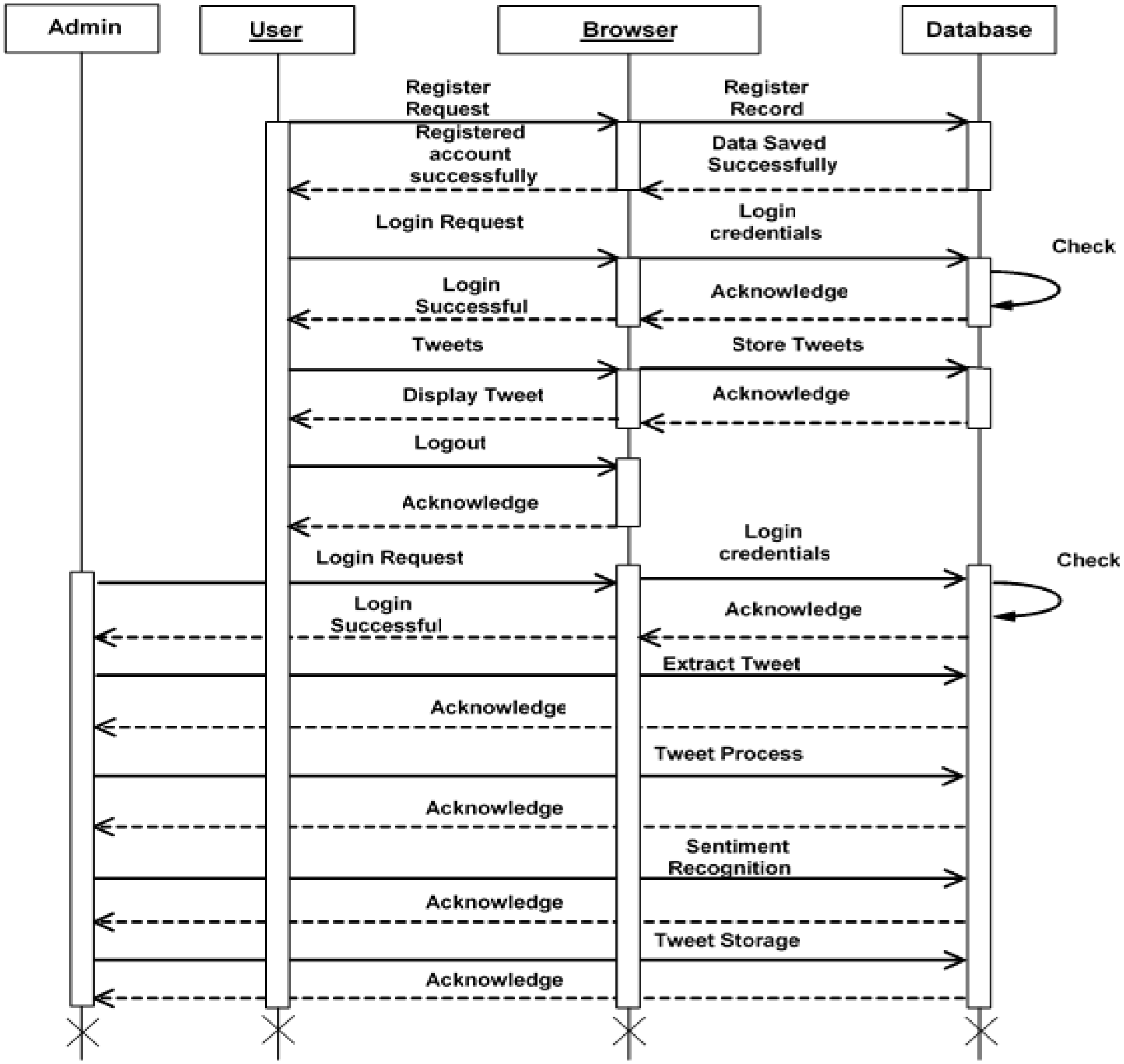


Fig. 5. 6. 2 Sequence Diagram

Class Diagram:

Class diagrams are the main building block of any object-oriented solution. It shows the classes in asystem, attributes, and operations of each class and the relationship between each class.

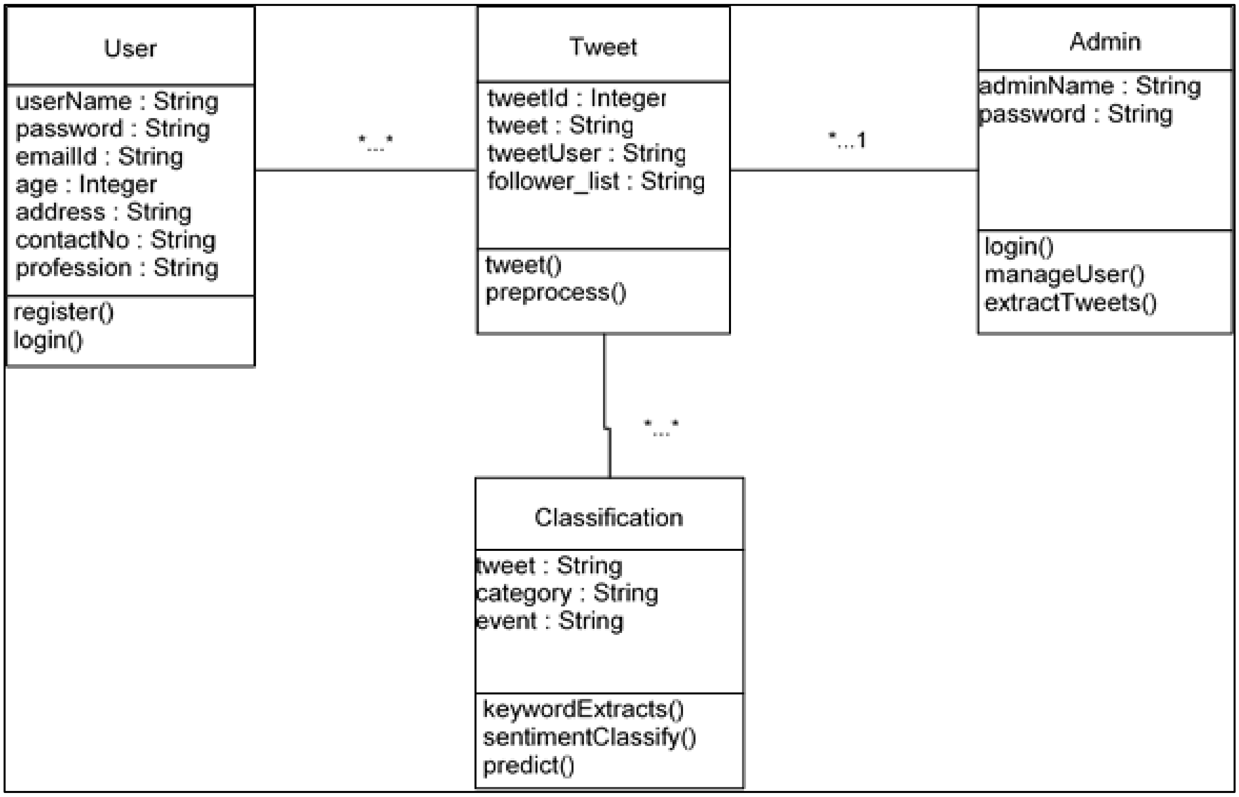


Fig. 5. 6. 3 Class Diagram

Activity Diagram:

Activity diagrams represent workflows in a graphical way. The activity diagram is:

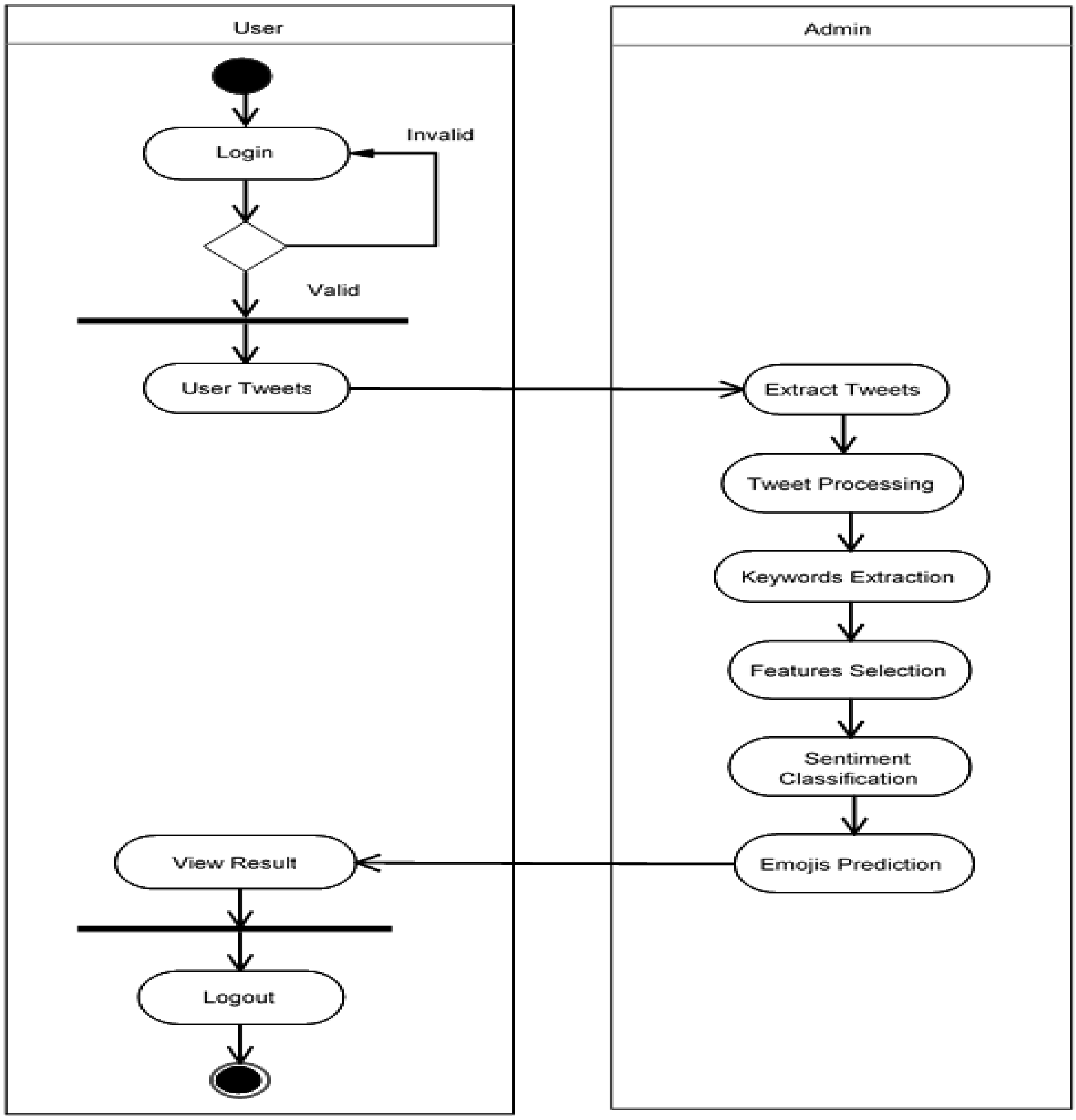


Fig. 5. 6. 4 Activity Diagram

Component Diagram:

A component diagram displays the structural relationship of components of a software system. These are mostly used when working with complex systems with many components. Components communicate with each other using interfaces. The interfaces are linked using connectors. The image below shows component diagram:

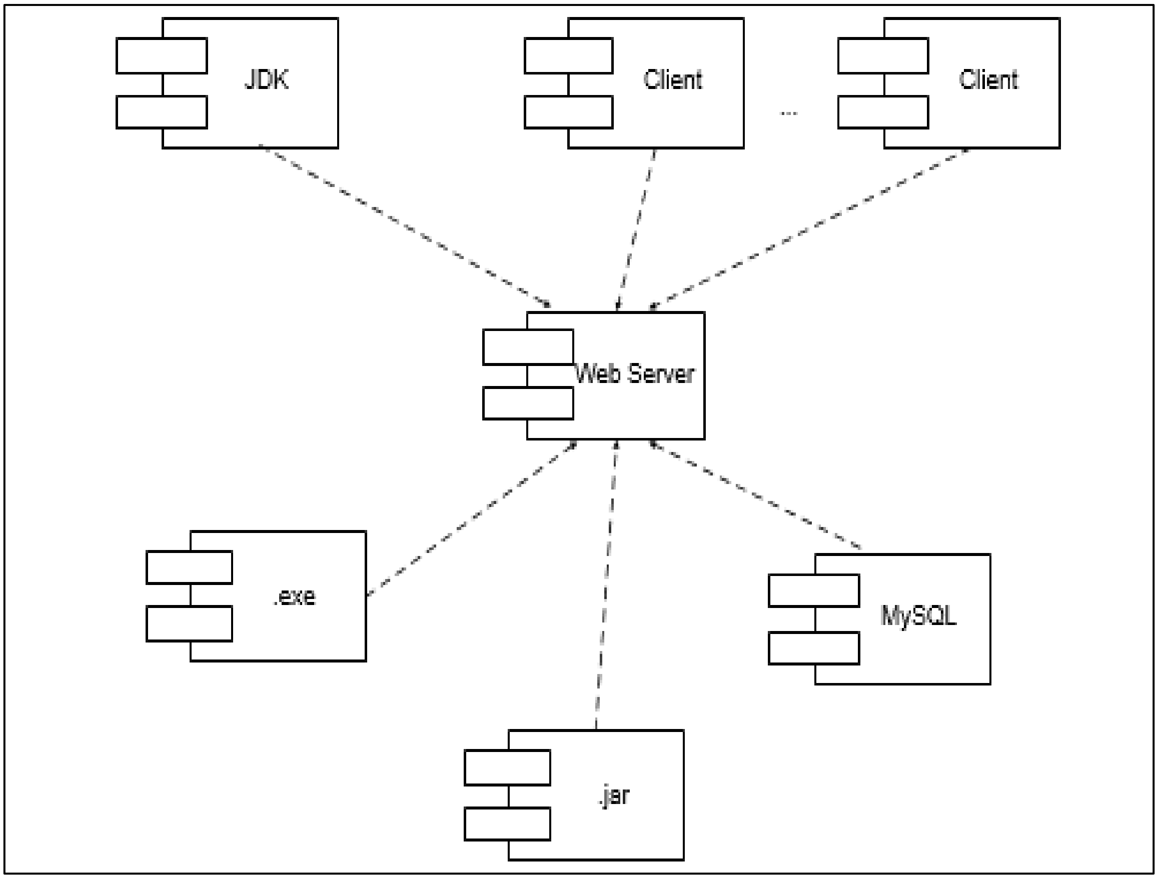


Fig. 5. 6. 5 Component Diagram

Deployment Diagram:

A deployment diagram shows the hardware of your system and the software in that hardware. Deployment diagrams are useful when your software solution is deployed across multiple machines with each having a unique configuration. Below is the deployment diagram:

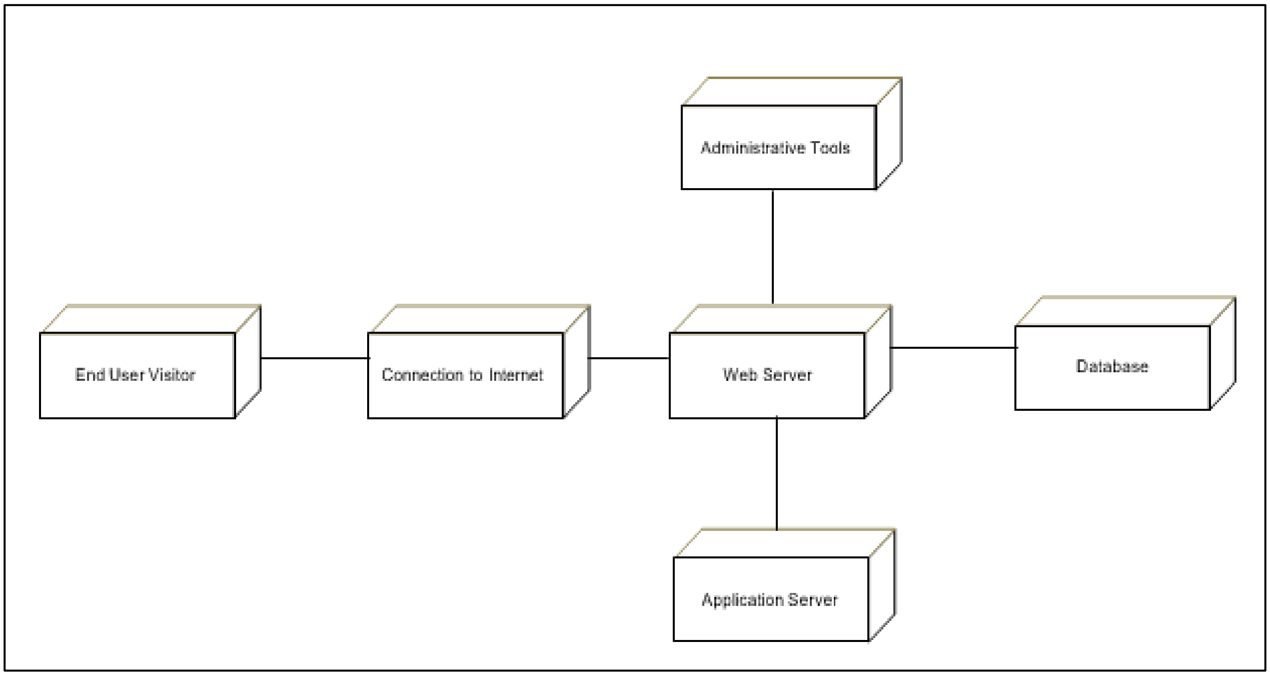


Fig. 5. 6. 6 Deployment Diagram

Chapter 6

SYSTEM IMPLEMENTATIONS

6.1 Important Libraries /Packages

1. NLTK

NLTK (Natural Language Toolkit) is a widely used open-source library for natural language processing (NLP) in Python. It provides various tools, resources, and algorithms for tasks like tokenization, stemming, lemmatization, part-of-speech tagging, parsing, sentiment analysis, and more. NLTK is designed to facilitate the development of NLP applications and research.

Key features and functionalities of NLTK include:

- Tokenization: NLTK provides tokenizers to split text into individual words or sentences. It supports different tokenization techniques, including word tokenization, sentence tokenization, and regular expression-based tokenization.

- Stemming and Lemmatization: NLTK offers stemming algorithms that reduce words to their root or stem form (e.g., converting "running" to "run"). It also provides lemmatization, which maps words to their base or dictionary form (e.g., converting "better" to "good").

- Part-of-Speech Tagging: NLTK includes pre-trained models and algorithms for part-of-speech tagging, which assigns grammatical tags to words in a sentence (e.g., noun, verb, adjective, etc.). It allows for analyzing the syntactic structure of text.

- Sentiment Analysis: NLTK offers resources and functionalities for performing sentiment analysis on text. It includes sentiment lexicons, pre-trained sentiment classifiers, and techniques for feature extraction and sentiment classification.

NLTK is widely used in academia and industry for NLP tasks, including text preprocessing, classification, and more. It provides a set of tools and resources for NLP practitioners and researchers, making it a valuable library for various NLP applications and projects in Python.

2. Text Blob

TextBlob is a Python library built on top of NLTK that provides an intuitive API for common natural language processing (NLP) tasks. It simplifies the implementation of various NLP operations, including sentiment analysis, part-of-speech tagging, noun phrase extraction, translation, and more.

Some key features and functionalities of TextBlob include:

- Sentiment Analysis: TextBlob offers a straightforward and easy-to-use sentiment analysis API. It provides pre-trained sentiment classifiers and lexicons to analyze the sentiment polarity (positive, negative, or neutral) of text. Sentiment scores can be obtained at the sentence or document level.
- Part-of-Speech Tagging: TextBlob performs part-of-speech (POS) tagging, assigning grammatical tags to words in a sentence. It uses the NLTK POS tagger under the hood and provides an intuitive interface to access the POS tags associated with each word.
- Noun Phrase Extraction: TextBlob includes a noun phrase extractor, allowing you to identify and extract noun phrases from text. Noun phrases are useful for extracting important information and understanding the structure of sentences.
- Tokenization and Word Inflection: TextBlob provides easy-to-use methods for tokenizing text into words or sentences. It also offers features for word inflection, allowing you to pluralize or singularize words, convert words to their base form, and perform basic text normalization.
- Language Translation: TextBlob integrates with the Google Translate API, allowing you to translate text between different languages with a simple API call. It supports translation from one language to another and provides language detection functionality.
- Spelling Correction: TextBlob includes a spelling correction feature that can automatically correct spelling mistakes in text. It uses a combination of statistical approaches and word frequency data to suggest corrections for misspelled words.

TextBlob aims to provide a user-friendly and intuitive interface for common NLP tasks, making it accessible for those without extensive NLP knowledge or experience. It combines the power of NLTK's underlying algorithms and resources with a simplified API, allowing developers to perform NLP operations quickly and easily.

3. SpaCy

SpaCy is a popular and efficient open-source library for natural language processing (NLP) in Python. It is designed to be fast, scalable, and production-ready, making it suitable for both research and industrial NLP applications. SpaCy provides a wide range of features and functionalities for various NLP tasks, including tokenization, part-of-speech tagging, dependency parsing, named entity recognition, and more.

Here are some key features and functionalities of SpaCy:

- Tokenization: SpaCy performs tokenization, splitting text into individual words or tokens, considering complex rules for handling punctuation, contractions, and specialized tokenization requirements.
- Part-of-Speech Tagging: SpaCy assigns part-of-speech tags to tokens, indicating their grammatical categories, such as noun, verb, adjective, and so on. It provides accurate and customizable POS tagging models that can be trained on specific domain data.
- Dependency Parsing: SpaCy performs dependency parsing, which analyzes the syntactic structure of a sentence and identifies the relationships between words. It constructs a dependency tree representing the grammatical relationships and dependencies between words in a sentence.
- Named Entity Recognition (NER): SpaCy offers robust named entity recognition capabilities to identify and classify named entities in text, including persons, organizations, locations, dates, and more. It provides pre-trained NER models that can be fine-tuned or trained on custom domain-specific data.
- Entity Linking: SpaCy supports entity linking, which involves linking named entities in text to corresponding entries in a knowledge base, such as Wikipedia or other domain-specific databases. This can help in disambiguating entities and providing additional contextual information.
- Word Vectors: SpaCy provides word vectors trained on large corpora, such as GloVe or fastText. These

word vectors capture semantic relationships between words and can be used for tasks like word similarity, word sense disambiguation, and word embedding-based operations.

- Text Classification: SpaCy supports text classification tasks, allowing you to train models for assigning pre-defined labels or categories to text documents. It provides a flexible framework for feature extraction, training models, and making predictions on new data.
- Customization and Extension: SpaCy is designed to be highly customizable and extensible. It allows users to train their own models on custom data, integrate external resources, add custom pipeline components, and fine-tune existing models for specific tasks.

SpaCy has gained popularity for its speed and efficiency, making it suitable for large-scale processing and real-time applications. It provides an easy-to-use API, comprehensive documentation, and a strong ecosystem, making it a valuable tool for various NLP tasks and projects.

6.2 Important Functions

Admin:

1. Extract Tweets:
   Admin takes the tweets which is send by the user. This is the first step which is carried out by the admin.
2. Tweet Processing:
   The admin uses the Natural Language Toolkit (NLTK) package, an open-source library for natural language processing (NPL).
   Pre-processing steps:

- Removing twitter handles
- Removing URLs
- Removing punctuations
- Removing handles
- Removing stopwords

- Stemming
- Tokenizing the string
- Lowercasing

3. Keywords Extraction:

   Keyword extraction is commonly used to extract key information from a series of sentence/phrase. Keyword extraction is an automated method of extracting the most relevant words and phrases from text input. It is a text analysis method that involves automatically extracting the most important words and expressions from the inputted tweet.

4. Features Selection:

   Feature selection is a method to identify a subset of features to achieve various goals: firstly, to reduce computational cost, secondly, to avoid over fitting, and thirdly, to enhance the classification accuracy of the model. Feature selection methods can be broadly divided into three categories, as filter methods, wrapper methods, and embedded methods.

5. Sentiment Classification:

   Sentiment classification is the automated process of identifying opinions in text and labeling them as positive, negative, or neutral, based on the emotions user express within them.

6. Emojis Prediction:

   Emojis are also known as most effective in expressing emotions in sentences. The admin can predict sentiment using emojis of text posted on webpage without labeling it manually.

User:

1. Login:

   A user can login to the webpage with the credentials created by the user. A user has its own username and password for login purpose.

2. User Tweets:

   User can input his/her tweets for getting the analysis of the sentiments.

3. View Result:

   User gets the result of the tweet such as positive, negative, and neutral.

4. Logout:

   User can logout from the webpage in this step after getting the analysis of the text.

6.3 Important Algorithms

Random Forest grows multiple decision trees which are merged for a more accurate prediction. The logic behind the Random Forest model is that multiple uncorrelated models (the individual decision trees) perform much better as a group than they do alone. When using Random Forest for classification, each tree gives a classification or a –vote. The forest chooses the classification with the majority of the –votes. The forest chooses the classification with the majority of the –votes. When using Random Forest for regression, the forest picks the average of the outputs of all trees.

Here is a general overview of how Random Forest can be applied to sentiment analysis:

- Dataset Preparation: Start by preparing a labeled dataset for sentiment analysis. This dataset should include text samples along with their corresponding sentiment labels (positive, negative, or neutral).

- Feature Extraction: Convert the text samples into numerical feature representations that can be used as input for the Random Forest algorithm. Common techniques for feature extraction in sentiment analysis include bag-of-words (BoW), TF-IDF (Term Frequency-Inverse Document Frequency), or word embeddings (such as Word2Vec or GloVe).

- Dataset Split: Split the dataset into training and testing sets. The training set will be used to train the Random Forest model, while the testing set will be used for evaluation.

- Random Forest Training: Train a Random Forest classifier on the training set. Random Forest consists of an ensemble of decision trees, where each tree is trained on a random subset of the features and data samples. The decision trees vote to determine the final sentiment prediction.

- Model Evaluation: Evaluate the trained Random Forest model using the testing set. Calculate performance metrics such as accuracy, precision, recall, or F1-score to assess the model's effectiveness in predicting sentiment.

- Prediction: Once the Random Forest model is trained and evaluated, it can be used to predict sentiment on new, unseen text samples. Apply the same feature extraction techniques used during training to convert the new text samples into numerical features, and then pass them through the trained Random Forest model for sentiment classification.

The key here lies in the fact that there is low (or no) correlation between the individual models—that is, between the decision trees that make up the larger Random Forest model. While individualdecision trees may produce errors, most of the group will be correct, thus moving the overall outcome in the right direction.

It is important to note that the performance of a Random Forest model for sentiment analysis can be influenced by various factors, such as the quality and size of the training dataset, the choice of features and their representation, as well as the hyperparameters of the Random Forest algorithm (e.g., the number of trees, maximum depth, etc.). Experimentation and fine-tuning of these parameters may be required to achieve optimal results.

Alternative Algorithms:

1. Naive Bayes: This algorithm is known to work well for many text classification problems and requires relatively few training examples. Naive Bayes is a commonly used algorithm for sentiment analysis due to its simplicity, speed, and effectiveness in text classification tasks. Naive Bayes assumes that features are conditionally independent given the class label. Different variations of Naive Bayes, such as Multinomial Naive Bayes or Bernoulli Naive Bayes, can be used based on the nature of the features.

2. Support Vector Machine: Like Naive Bayes classifiers, support vector classifiers also work well for text classification and require relative few training examples. It is a popular machine learning algorithm that can be effectively used for sentiment analysis tasks. SVM has been widely used for sentiment analysis due to its ability to handle high-dimensional feature spaces, capture complex decision boundaries, and effectively handle small to medium-sized datasets. It can handle both linearly separable and non-linearly separable sentiment classification problems by utilizing different kernel functions.

3. Decision Tree: Decision Trees often do a good job of learning to classify and have the additional property of producing easily explainable results in the form of decision trees. Decision Tree is another machine learning algorithm that can be used for sentiment analysis tasks. Decision Trees are known for their interpretability and ability to capture complex decision rules. However, they can suffer from overfitting if not properly controlled. Techniques such as pruning, setting a maximum depth for the tree, or using ensemble methods like Random Forest can help mitigate overfitting and improve generalization.

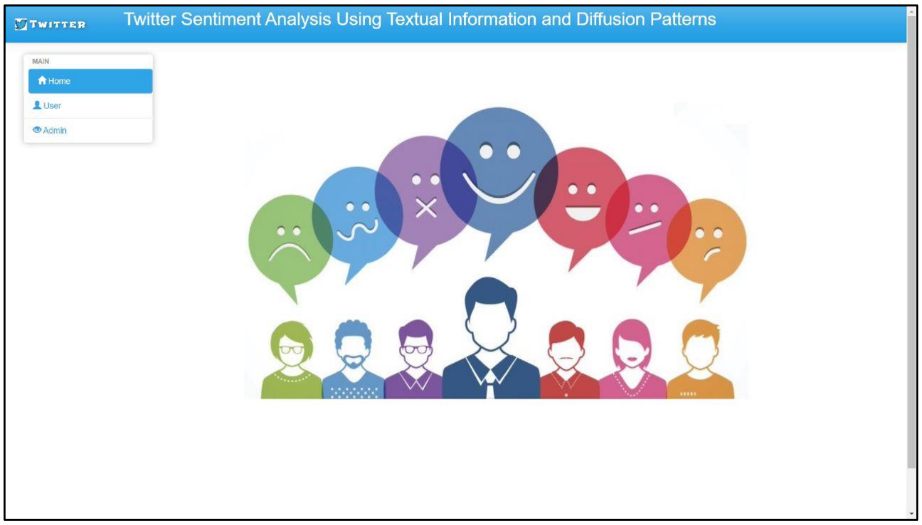
4. XGBoost: This algorithm uses a set of different decision trees known as a random forest. It is known to be both fast and often achieves very high accuracy. However, it is not as interpretable as a simple decision tree. It (Extreme Gradient Boosting) is a powerful machine learning algorithm that can be used for sentiment analysis tasks. It is an optimized implementation of gradient boosting that combines multiple weak prediction models, such as decision trees, to create a strong ensemble model. XGBoost has gained popularity due to its efficiency, scalability, and high performance. XGBoost offers several hyperparameters that can be tuned to optimize performance, such as the learning rate, number of trees (boosting rounds), maximum depth of trees, and regularization parameters. Hyperparameter tuning,

along with feature selection and preprocessing techniques, can significantly impact the performance of the XGBoost model in sentiment analysis tasks.

5. k-Nearest Neighbors: This algorithm works by finding the training examples closest to the test example. It is a popular machine learning algorithm that can be used for sentiment analysis tasks. KNN is a non-parametric algorithm that classifies a sample based on the majority class of its K nearest neighbors in the feature space. It is a simple and intuitive algorithm, but it can be sensitive to the choice of K (the number of neighbors) and the distance metric. It is also computationally intensive, especially for large datasets, as it requires calculating distances between each sample and all training samples during prediction. Experimentation with different preprocessing techniques, feature extraction methods, the number of neighbors (K), and distance metrics can help optimize the performance of the KNN model for sentiment analysis tasks.

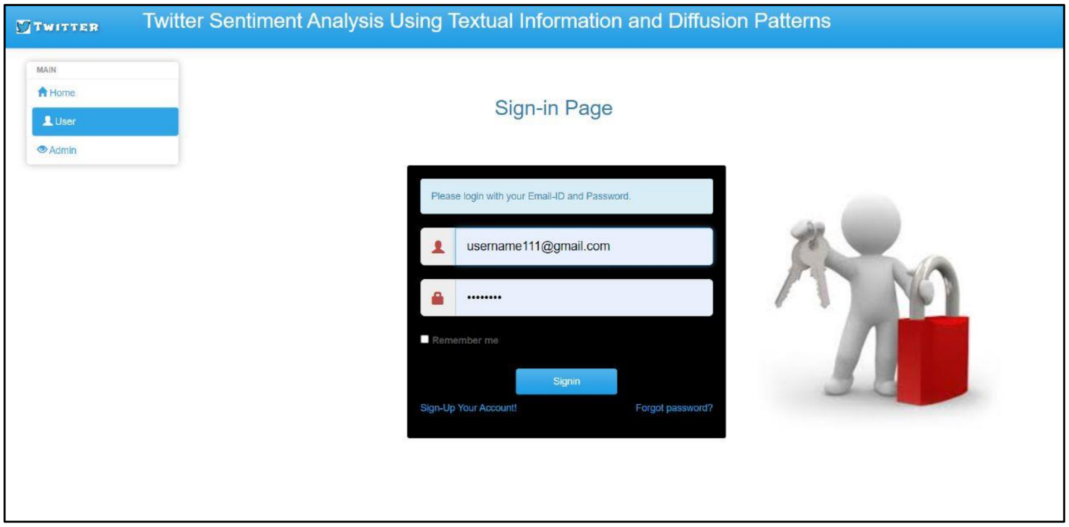
6. Logistic Regression: It is a classification that serves to solve the binary classification problem. The result is usually defined as 0 or 1 in the models with a double situation. It is a commonly used algorithm for sentiment analysis tasks. Despite its name, it is a classification algorithm that works well for binary classification problems like sentiment analysis (positive or negative sentiment). Logistic Regression is known for its simplicity, interpretability, and efficiency. It can handle both linear and non-linear relationships between the features and the target variable. However, it may not capture complex interactions among features as effectively as other algorithms like Random Forest or Deep Learning models.

6.4 Graphics User Interface Screenshots



Twitter Sentiment Analysis Using Textual Information and Diffusion Patterns

Fig. 6. 4. 1 Home page



Twitter Sentiment Analysis Using Textual Information and Diffusion Patterns

Sign-in Page

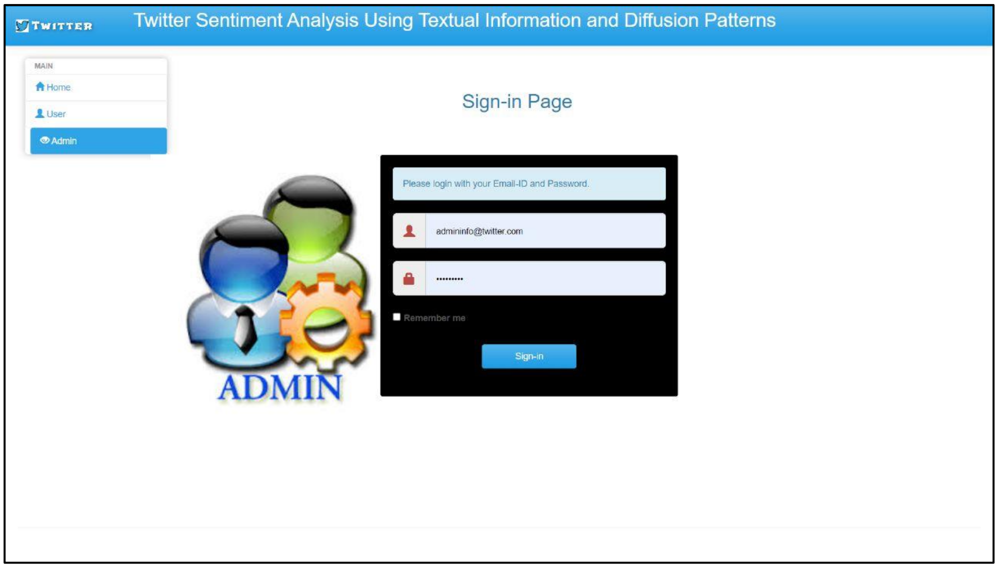
Please login with your Email-ID and Password.

username111@gmail.com

••••••••

□ Remember me

Fig. 6.4.2 User Sign in Page



Twitter Sentiment Analysis Using Textual Information and Diffusion Patterns

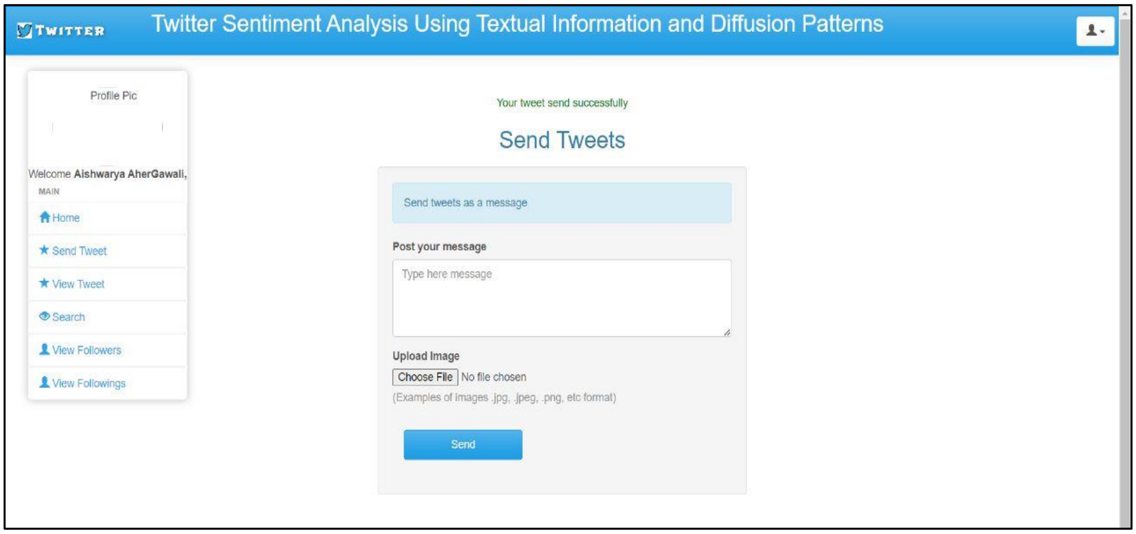
Sign-in Page

Please login with your Email-ID and Password.

admininfo@twitter.com

....

Fig. 6.4.3 Admin Sign-in Page



Twitter Sentiment Analysis Using Textual Information and Diffusion Patterns

Welcome Aishwarya AherGawali,

Your tweet send successfully

Send Tweets

Send tweets as a message

Post your message

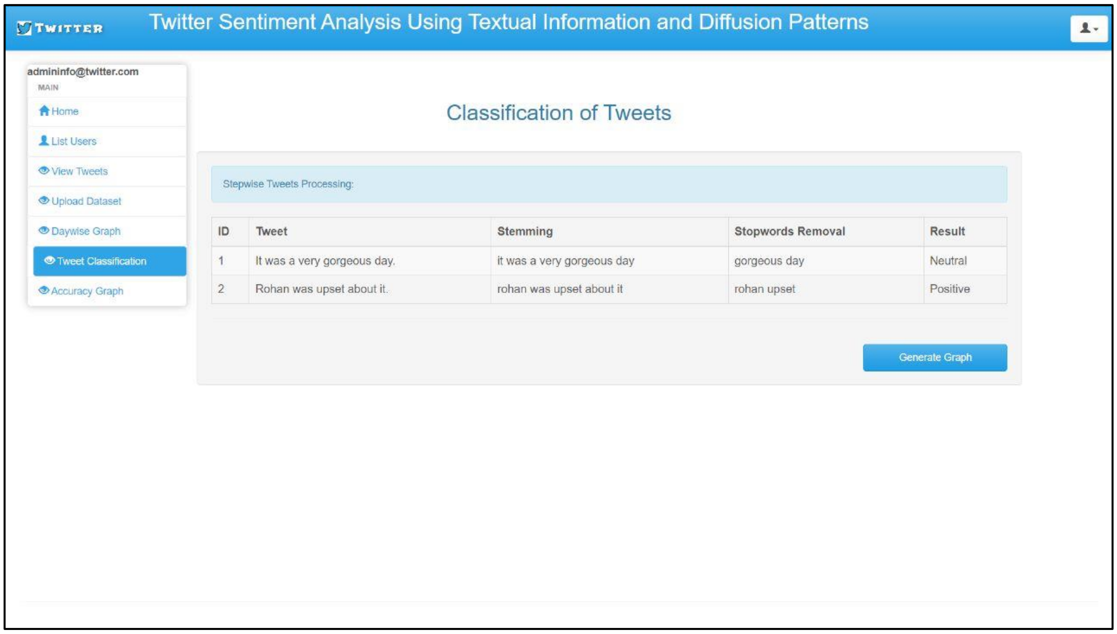
Type here message

Upload Image

No file chosen

(Examples of images .jpg, .jpeg, .png, etc format)

Fig. 6.4.4. Send Tweets Page



Classification of Tweets

Stepwise Tweets Processing:

ID	Tweet	Stemming	Stopwords Removal	Result
1	It was a very gorgeous day.	it was a very gorgeous day	gorgeous day	Neutral
2	Rohan was upset about it.	rohan was upset about it	rohan upset	Positive

Fig. 6.4.5 Classification of Tweets

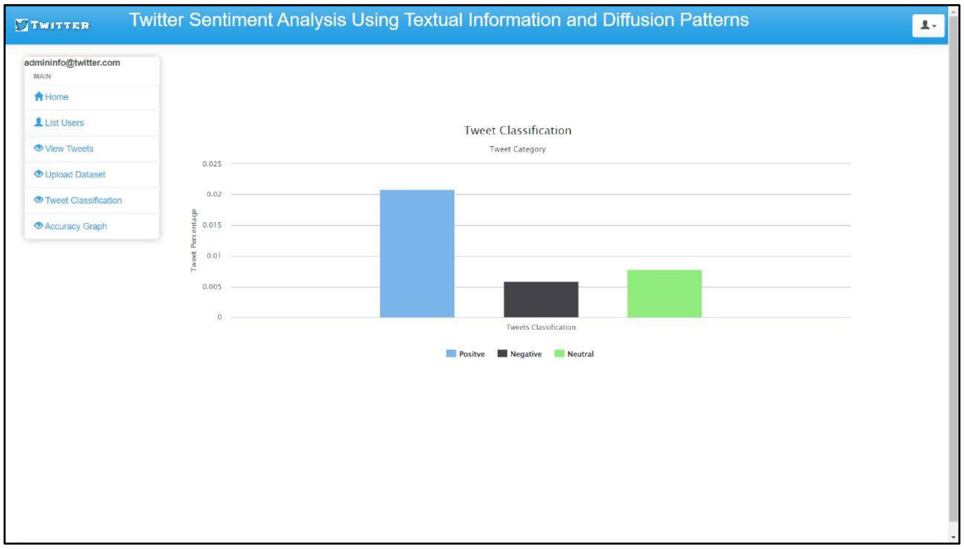
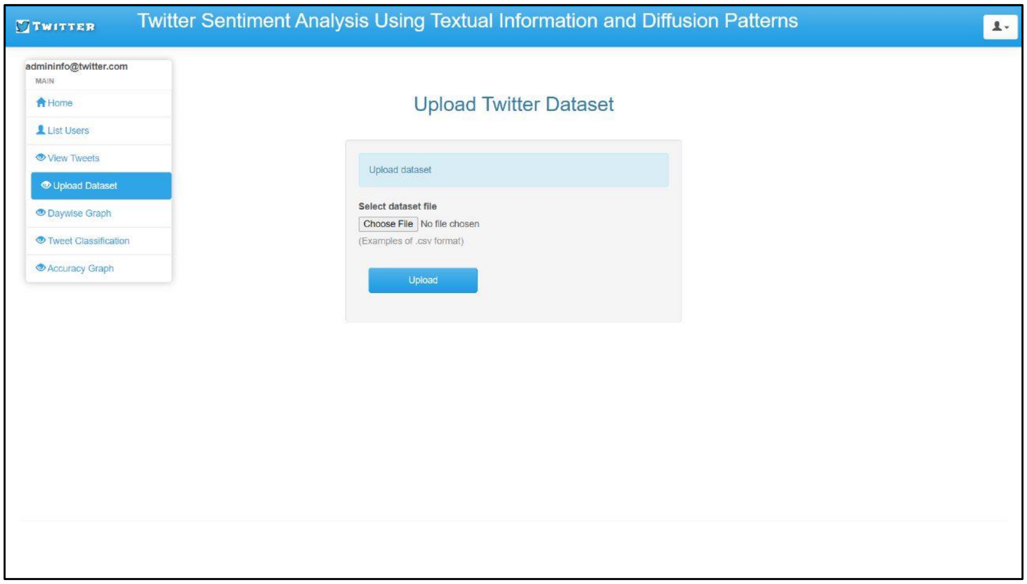


Fig. 6.4.6 Tweet Classification Graph



Upload Twitter Dataset

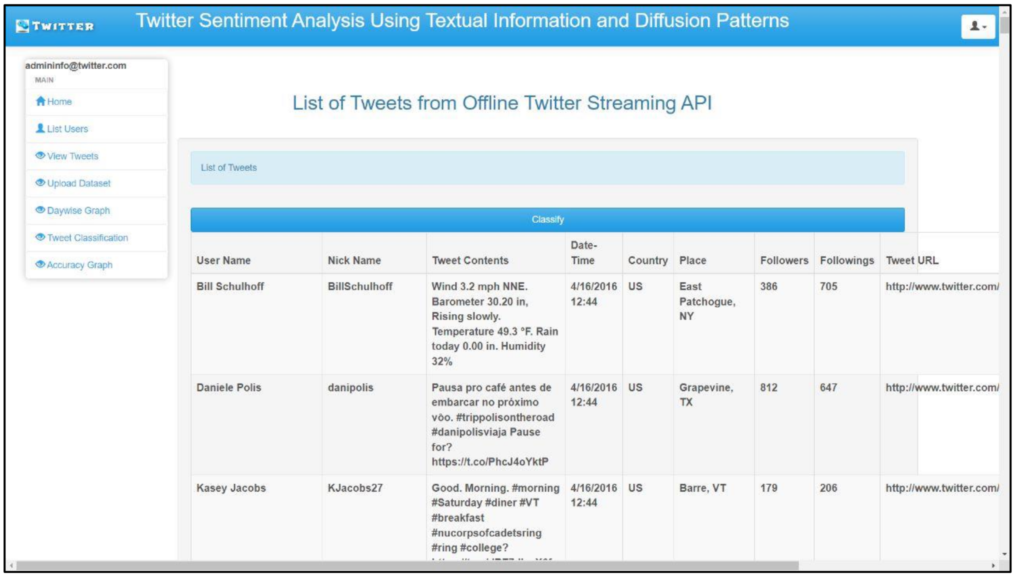
Upload dataset

Select dataset file

Choose File No file chosen

(Examples of .csv format)

Fig. 6.4.7 Twitter Dataset Upload Page



List of Tweets from Offline Twitter Streaming API

List of Tweets

User Name	Nick Name	Tweet Contents	Date-Time	Country	Place	Followers	Followings	Tweet URL
Bill Schulhoff	BillSchulhoff	Wind 3.2 mph NNE. Barometer 30.20 in, Rising slowly. Temperature 49.3 °F. Rain today 0.00 in. Humidity 32%	4/16/2016 12:44	US	East Patchogue, NY	386	705	http://www.twitter.com/
Daniele Polis	danipolis	Pausa pro café antes de embarcar no próximo vôo. #trippolisontheroad #danipolisviaja Pause for? https://t.co/PhcJ4oYktP	4/16/2016 12:44	US	Grapevine, TX	812	647	http://www.twitter.com/
Kasey Jacobs	KJacobs27	Good. Morning. #morning #Saturday #diner #VT #breakfast #nucorpsofcadetsring #ring #college?	4/16/2016 12:44	US	Barre, VT	179	206	http://www.twitter.com/

Fig. 6.4.8 List of Tweets from Offline Twitter Streaming API

Chapter 7

SYSTEM TESTING

Software Testing:

Software testing is known as a process for validating and verifying the working of a software/application. It makes sure that the software is working without any errors, bugs, or any other issues and gives the expected output to the user. The software testing process does not limit to finding faults in the present software but also finding measures to upgrade the software in various factors such as efficiency, usability, and accuracy. So, to test software the software testing provides a particular format called a Test Case. A test case is a defined format for software testing required to check if a particular application/software is working or not. A test case consists of a certain set of conditions that need to be checked to test an application or software i.e., in more simple terms when conditions are checked it checks if the resultant output meets with the expected output or not. A test case consists of various parameters such as Id, condition, steps, input, expected result, result, status, and remarks.

Parameters of a Test Case:

- **Module Name:** Subject or title that defines the functionality of the test.
- **Test Case Id**: A unique identifier assigned to every single condition in a test case.
- **Tester Name:** The name of the person who would be carrying out the test.
- **Test scenario:** The test scenario provides a brief description to the tester, as in providing a small overview to know about what needs to be performed and the small features, and components of the test.
- **Test Case Description:** The condition required to be checked for a given software. for e.g. Check if only numbers validation is working or not for an age input box.
- **Test Steps:** Steps to be performed for the checking of the condition.
- **Prerequisite:** The conditions required to be fulfilled before the start of the test process.
- **Test Priority:** As the name suggests gives the priority to the test cases as in which had to be performed first, or are more important and which could be performed later.

- **Test Data:** The inputs to be taken while checking for the conditions.
- **Test Expected Result:** The output which should be expected at the end of the test.
- **Test parameters**: Parameters assigned to a particular test case.
- **Actual Result:** The output that is displayed at the end.
- **Environment Information:** operating system, security information, software version.
- **Status**: The status of tests such as pass, fail, NA, etc.
- **Comments**: Remarks on the test regarding the test for the betterment of the software.

Table 3: Software Testing

Test Case Type	Description	Test Step	Expected Result	Status
Login Page	Input username and password	Password up to 8 letters including special characters	Password should be verified	Pass
Login page	Input username and password	Password more than 8 characters	Invalid password	Fail
Dataset	Upload dataset	Should be a .csv file	Uploaded successfully	Pass
Dataset	Upload Dataset	Other than .csv file	Unsuccessful	Fail
User Sign up	Set password	Password and confirm password should match	Password mismatch	Fail
User sign up	Set password	Password and confirm password should match	Password match	Pass
Admin login page	Input username and password	Username should be valid	Username invalid	Fail
Admin login page	Input username and password	Username should be valid	Username valid	Pass
Change password	Enter New password	Password up to 8 letters includingspecial characters	New and old password same	Fail
Change password	Enter New password	Password up to 8 letters includingspecial characters	New password valid	Pass

Chapter 8

EXPERIMENTAL RESULTS

The section shows overall accuracy of Naïve Bayes and Random Forest algorithm. So, this works gives better sentiment analysis results compare to existing method.

The experimental result evaluation, we have notation as follows:

- TP: True positive (correctly predicted number of instance)
- FP: False positive (incorrectly predicted number of instance),
- TN: True negative (correctly predicted the number of instances as not required)
- FN false negative (incorrectly predicted the number of instances as not required),

Based on this parameter, we can calculate four measurements:

- Accuracy = TP+TN÷TP+FP+TN+FN
- Precision = TP ÷TP+FP
- Recall= TP÷TP+FN
- F1-Measure = 2×Precision×Recall ÷Precision+ Recall.

Table 4: Accuracy Table

Sr. No.	**Existing System**	**Proposed System**
Precision	60.2%	65.4 %
Recall	85.5%	89.7%
Accuracy	72%	86%

After performing sentiment analysis on a collection of tweets related to a specific topic or hashtag. The sentiment analysis algorithm assigns sentiment labels to each tweet, indicating whether the sentiment expressed in the tweet is positive negative or neutral.

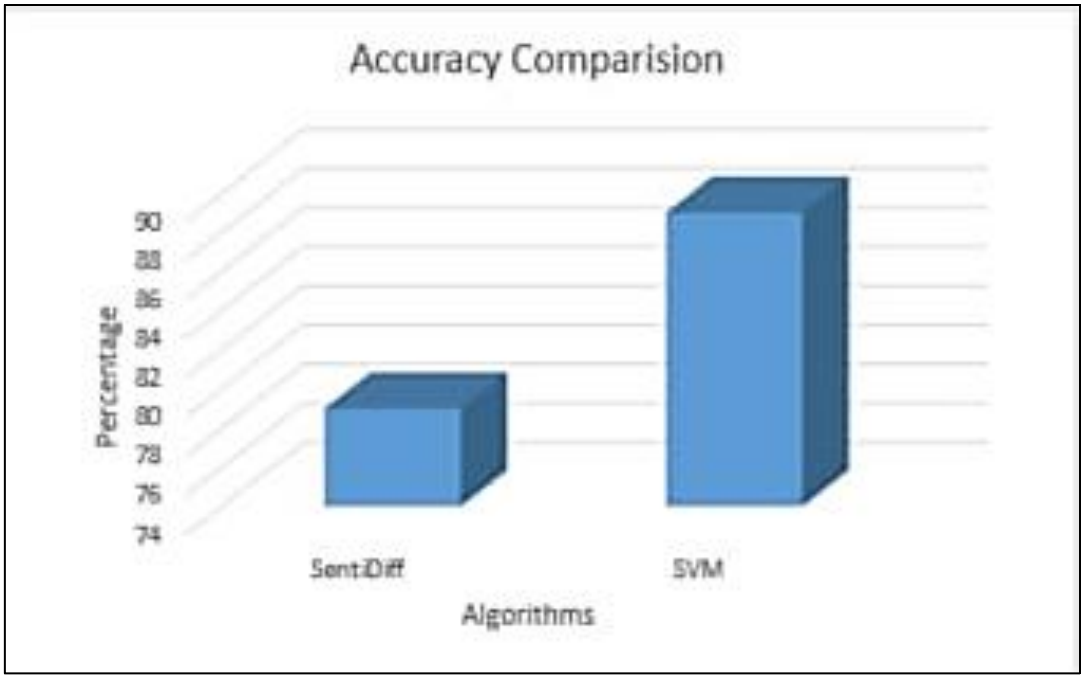
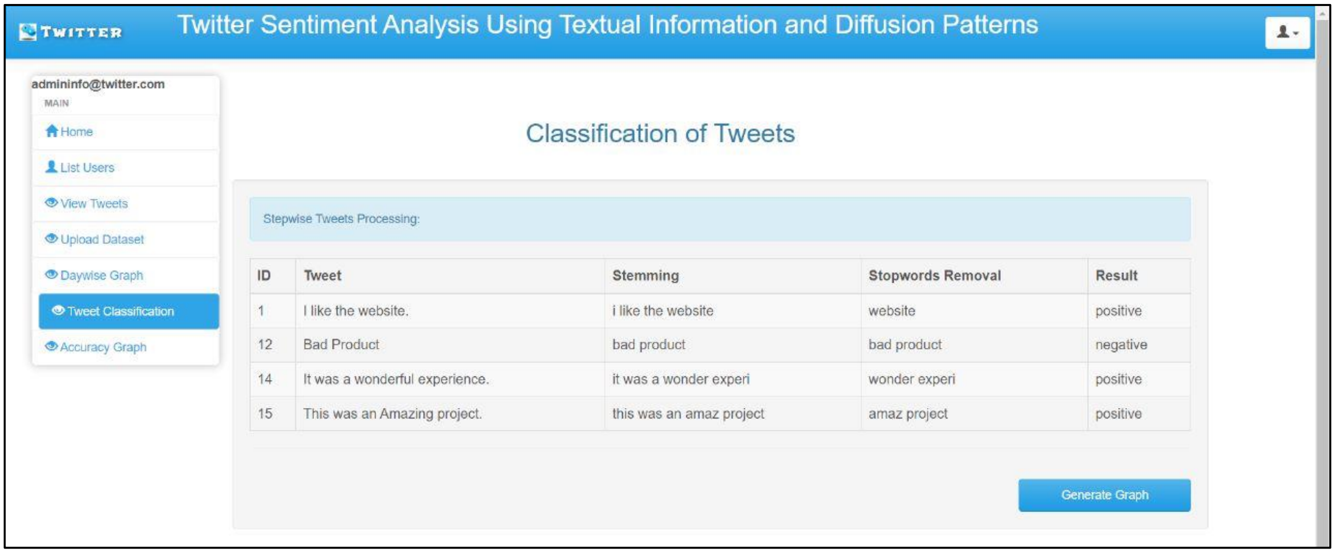


Fig. 8.1 Accuracy Result



Classification of Tweets

Stepwise Tweets Processing:

ID	Tweet	Stemming	Stopwords Removal	Result
1	I like the website.	i like the website	website	positive
12	Bad Product	bad product	bad product	negative
14	It was a wonderful experience.	it was a wonder experi	wonder experi	positive
15	This was an Amazing project.	this was an amaz project	amaz project	positive

Fig. 8.2 Classification of Tweets

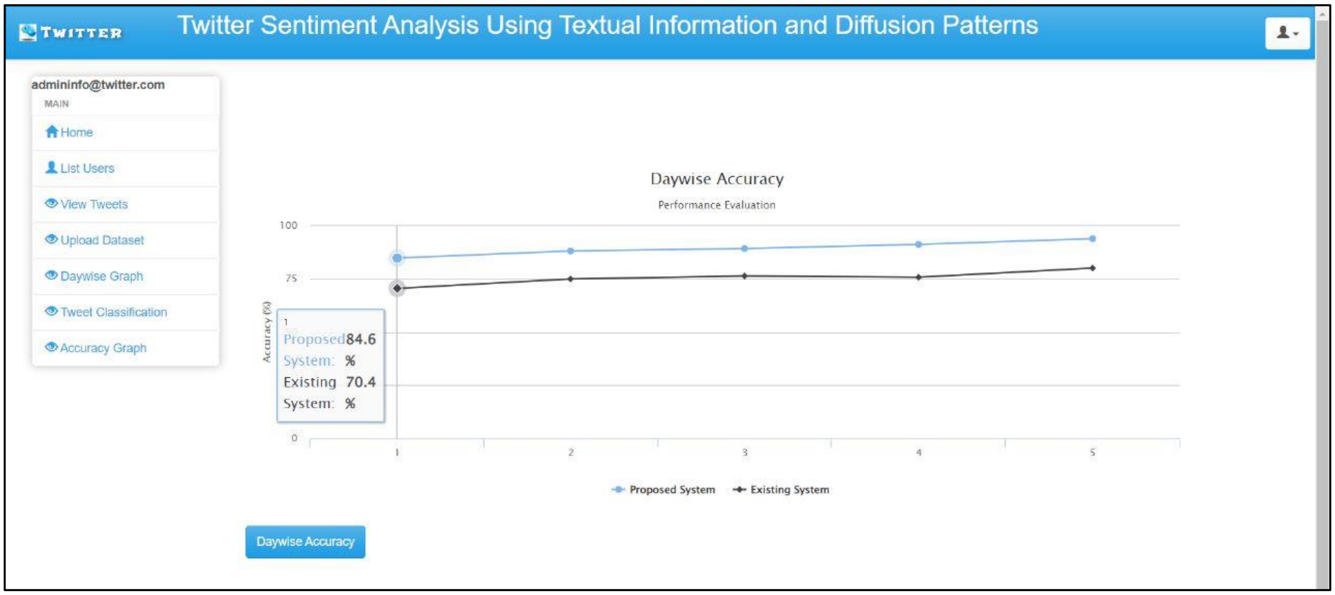


Fig. 8.3 Day wise Accuracy

Chapter 9

CONCLUSION

Polarity of mining sentiments expressed in Twitter messages is a significant and challenging task. Most existing Twitter sentiment analysis solutions consider only the textual information of Twitter messages and cannot achieve satisfactory performance due to the unique characteristics of Twitter messages. Although recent studies have shown that patterns of feeling diffusion are closely related to the polarities of Twitter messages, existing approaches are essentially based only on textual information from Twitter messages, but ignore the dissemination of information about feelings. Inspired by the recent work on the fusion of knowledge of multiple domains, take a first step towards combining textual information and spreading feelings to get a better performance of Twitter 's sentiment analysis.

In conclusion, this project report highlights the significance of sentiment analysis in understanding and analyzing textual data. We have successfully developed and implemented a sentiment analysis system using a comprehensive methodology. Our findings demonstrate the effectiveness of different models, preprocessing techniques, and feature extraction methods in accurately classifying sentiment. The insights gained from this project can be applied to various domains, benefiting businesses, governments, and individuals alike. The report concludes with suggestions for future research and enhancements in sentiment analysis techniques to further improve accuracy and efficiency.

Overall, our sentiment analysis project contributes to the growing field of opinion mining and provides a foundation for further exploration and application of sentiment analysis techniques in real-world scenarios.

Chapter 10

FUTURE SCOPE

1. **Live tweet classification:**

Live tweet classification refers to the real-time analysis and classification of tweets based on their sentiment or emotion. While sentiment analysis has traditionally focused on analyzing pre-existing textual data, the future scope of sentiment analysis includes the ability to classify tweets as they are generated in real-time. Here are some potential applications and benefits of live tweet classification:

- Real-time Event Monitoring: Live tweet classification can be used to monitor public sentiment and reactions during live events, such as conferences, sports matches, or political debates. By analyzing and classifying tweets in real-time, event organizers, media outlets, or marketers can understand the audience's sentiment and adjust their strategies accordingly.
- Social Media Advertising: Live tweet classification can enhance targeted advertising on social media platforms. By classifying tweets in real-time, advertisers can identify users expressing sentiments that align with their target audience and deliver personalized ads that resonate with their interests and emotions.
- Trend Identification: Live tweet classification enables the identification of emerging trends and topics on social media platforms. By analyzing and categorizing real-time tweets, businesses, media organizations, or researchers can identify popular discussions, sentiment shifts, or emerging issues, allowing them to adapt their strategies and capitalize on relevant trends.
- Customer Support: Live tweet classification can enhance customer support on social media platforms. By automatically analyzing and categorizing tweets mentioning support-related keywords or issues, businesses can identify and prioritize customer inquiries, ensuring timely and appropriate responses.
- Real-time Public Opinion Analysis: Live tweet classification can aid in understanding public sentiment and opinions on current events, political issues, or social topics. By analyzing and categorizing real-time tweets, policymakers, researchers, or media outlets can gain insights into

public opinion, assess the impact of their actions or policies, and make informed decisions accordingly.

- Personalized Real-time Recommendations: Live tweet classification can enable platforms to deliver personalized recommendations and content to users in real-time. By understanding the sentiment and interests expressed in live tweets, platforms can provide tailored suggestions, news updates, or relevant information, enhancing user engagement and satisfaction.

2. **Emoji predictions:**

   Emoji prediction as a future scope of Twitter sentiment analysis has the potential to enhance the accuracy and depth of understanding of user sentiments. Emojis have become an integral part of online communication, and they can convey emotions and sentiments in a concise and expressive manner. By incorporating emoji prediction into sentiment analysis, Twitter can gain valuable insights into user sentiment beyond the limitations of text analysis alone.

3. **To identify ironical statement:**

   Identifying ironic statements as a future scope of sentiment analysis is indeed a promising area of research. Irony is a complex linguistic phenomenon that involves conveying the opposite of what is meant, often for humorous or sarcastic effect. Detecting irony in text can be challenging because it requires understanding the speaker's intent and recognizing the disparity between the literal and intended meaning.

Chapter 11

REFERENCES

[1] S. Symeonidis, D. Effrosynidis, and A. Arampatzis, ―A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis, ‖ Expert Systems with Applications, 2018

[2] J. Zhao and X. Gui, ―Comparison research on text pre-processing methods on twitter sentiment analysis, ‖ IEEE Access, vol. 5, pp. 2870–2879, 2017.

[3] X. Zhang, D.-D. Han, R. Yang, and Z. Zhang, ―Users participation and social influence during information spreading on twitter, ‖ PloS one, vol. 12, no. 9, p. e0183290, 2017.

[4] K. Schouten and F. Frasincar, ―Survey on aspect-level sentiment analysis, ‖ IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 3, pp. 813–830, 2016.

[5] S. Tsugawa and H. Ohsaki, ―Negative messages spread rapidly and widely on social media, ‖ in Proceedings of the 2015 ACM on Conference on Online Social Networks. ACM, 2015, pp. 151–160.

[6] S. M. Mohammad and S. Kiritchenko, ―Using Hashtags to Capture Fine Emotion Categories from Tweets, ‖ Computational Intelligence, vol. 31, no. 2, pp. 301–326, 2015.

[7] B. Plank and D. Hovy, ―Personality Traits on Twitter —or— How to Get 1,500 Personality Tests in a Week‖, Proc. of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2015, pp. 92–98.

[8] S. M. Mohammad, X. Zhu, S. Kiritchenko, and J. Martin, ―Sentiment, emotion, purpose, and style in electoral tweets‖ Information Processing and Management, vol. 51, no. 4, pp. 480–499, 2015.

[9] J. Bollen, H. Mao, and X.-J. Zeng, ―Twitter mood predicts the stock market, ‖ J. of Computational Science, vol. 2, no. 1, pp. 1–8, 2011.

[10] S. M. Mohammad and P. D. Turney, ―Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon, ‖ in Proc. of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text. ACL, 2010, pp. 26–34.

Chapter 12

APPENDIX

A. **Project Review I Remark**
B. **Project Review II Remark**
C. **Progress Report I**
D. **Project Review III Remark**
E. **Project Review IV Remark**
F. **Progress Report II**
G. **Journal/conference Publications**
H. **Participation Certificates**
I. **Plagiarism Report**
J. **Rubrics**
K. **Critical Thinking Questions**