Minor Project Report

On

Sentiment analysis

Submitted in partial fulfilment of requirements for the award of the Degree of

Bachelor of Technology

In

Information Technology

Submitted By

Shwetank Chaudhary

(07811503120)

Under the guidance of

Dr. Surinder Kaur

Department of Information Technology
Bharati Vidyapeeth's College of Engineering, New Delhi – 110063, INDIA

December 2023

CANDIDATE'S DECLARATION

It is hereby certified that the work which is being presented in the B.Tech Minor Project Report entitled **"Sentiment Analysis"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** and submitted in the **Department of Information Technology** of **BHARATI VIDYAPEETH'S COLLEGE OF ENGINEERING, New Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi)** is an authentic record of my own work carried out during a period from **August – December 2023** under the guidance of **, Dr. Surinder Kaur**

The matter presented in the B. Tech Minor Project Report has not been submitted by me for the award of any other degree of this or any other Institute.

Student Name: Shwetank Chaudhary
Enrollment No: 07811503120

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Dr. Surinder Kaur	**Prof. (Dr) Prakhar Priyadarshi**
Assistant professor	**HOD, IT**

ACKNOWLEDGEMENT

We express our deep gratitude to **Dr. Surinder Kaur,** Assistant Professor, Department of Information Technology for her valuable guidance and suggestions throughout my project work.

We would like to extend sincere thanks to the Head of Department, **Prof. (Dr.) Prakhar Priyadarshi** for his time-to-time suggestions to complete my project work. We are also thankful to **Prof. (Dr.) Dharmendra Saini,** Principal for providing me the facilities to carry out my project work.

Shwetank Chaudhary

(07811503120)

Abstract

Sentiment analysis, a branch of natural language processing, holds a pivotal role in the era of Big Data, providing a mechanism to distil valuable insights from the vast pool of textual information generated online. This abstract explores the evolving scope of sentiment analysis, encompassing its applications, challenges, and the transformative potential it carries across diverse domains.

Sentiment analysis involves employing computational algorithms to decipher the emotional tone expressed in text, enabling the classification of sentiments as positive, negative, or neutral. The scope of sentiment analysis has broadened significantly, finding applications in fields such as marketing, customer feedback analysis, social media monitoring, political discourse, and product reviews. As organizations increasingly recognize the importance of understanding public opinion, sentiment analysis has become an invaluable tool for decision-making and strategy formulation.

The abstract also addresses the scope of sentiment analysis in multilingual and cross-cultural contexts. With the proliferation of global communication, the ability to analyse sentiments across various languages and cultural nuances enhances the applicability of sentiment analysis in an interconnected world.

While sentiment analysis offers substantial benefits, it is not without challenges. Contextual understanding, handling sarcasm, and addressing biases in training data are persistent issues. Researchers and practitioners are actively exploring innovative solutions, leveraging advancements in machine learning and deep learning to enhance the accuracy and robustness of sentiment analysis models.

Furthermore, the abstract explores the ethical considerations surrounding sentiment analysis, emphasizing the need for responsible AI practices. Issues related to privacy, data security, and potential misuse of sentiment analysis outputs underscore the importance of ethical guidelines and regulatory frameworks.

In conclusion, this abstract underscore the expanding scope of sentiment analysis as a powerful tool in the extraction of valuable insights from the deluge of textual data. As technology continues to advance, and the scope of textual data expands, sentiment analysis stands poised to play an increasingly integral role in shaping informed decision-making processes across diverse sectors, ultimately contributing to a deeper understanding of human sentiments in the digital age.

Table of Contents

CONTENTS

Chapter 1 :

Introduction

In the era of information abundance and digital communication, the analysis of sentiments expressed in textual data has emerged as a critical tool for extracting meaningful insights and understanding the pulse of individuals and communities. This chapter sets the stage for our comprehensive exploration of sentiment analysis, delving into the background, significance, and objectives of this report.

1.1 Background

The proliferation of online content through social media, product reviews, news articles, and customer feedback has given rise to an unprecedented volume of textual data. Within this sea of information, understanding the sentiments conveyed by individuals has become crucial for businesses, researchers, and decision-makers. Sentiment analysis, also known as opinion mining, employs natural language processing techniques and machine learning algorithms to discern sentiments—be they positive, negative, or neutral—from textual content.

1.2 Significance of Sentiment Analysis

Understanding the sentiments of users and consumers is instrumental in shaping strategic decisions across various sectors. Businesses utilize sentiment analysis to gauge customer satisfaction, monitor brand perception, and identify areas for improvement. In the realm of social media, sentiment analysis helps track public opinion, detect emerging trends, and assess the impact of events. Moreover, sentiment analysis has applications in political discourse, market research, and beyond, making it a versatile and powerful tool for extracting actionable insights.

1.3 Objectives of the Report

This report aims to provide a comprehensive understanding of sentiment analysis, spanning its methodologies, applications, challenges, and future directions. Key objectives include:

1. Methodological Exploration: Uncover the underlying techniques and methodologies employed in sentiment analysis, ranging from rule-based systems to advanced machine learning models.

2. Application Landscape: Investigate the diverse applications of sentiment analysis across industries, including marketing, customer service, social media, politics, and more.

3. Challenges and Limitations: Examine the inherent challenges and limitations in sentiment analysis, such as handling context, mitigating biases, and addressing multilingual complexities.

4. Technological Advancements: Explore the latest technological advancements and trends in sentiment analysis, including the integration of deep learning and natural language processing innovations.

1.4 Structure of the Report

The subsequent chapters of this report will delve into each of these objectives, offering a comprehensive and nuanced exploration of sentiment analysis. Chapter 2 will provide an in-depth review of methodologies, Chapter 3 will focus on applications, Chapter 4 will tackle challenges, and Chapter 5 will discuss technological advancements. The final chapter will conclude our exploration, summarizing key findings and providing insights into the future of sentiment analysis.

In conclusion, this chapter establishes the foundation for a thorough investigation into sentiment analysis—a field poised at the intersection of linguistics, machine learning, and societal dynamics. The subsequent chapters will navigate through the intricacies of sentiment analysis, unravelling its significance and impact on diverse facets of our interconnected-world .

Chapter 2:

Methodologies in Sentiment Analysis

Sentiment analysis encompasses a spectrum of methodologies, each designed to decode the emotional undercurrents embedded within textual data. This chapter delves into the diverse approaches employed in sentiment analysis, ranging from traditional rule-based systems to advanced machine learning models.

2.1 Rule-Based Systems

Rule-based systems rely on predefined linguistic rules to categorize text into positive, negative, or neutral sentiments. These systems leverage lexicons, dictionaries, and linguistic patterns to match and assign sentiments based on the presence of specific words or phrases. While simple in concept, rule-based systems often struggle with context sensitivity and may not capture the nuances of language effectively.

2.2 Machine Learning Approaches

2.2.1 Supervised Learning

Supervised learning involves training a model on labelled datasets, where each piece of text is associated with a predefined sentiment category. Common algorithms include Support Vector Machines (SVM), Naive Bayes, and Decision Trees. Supervised learning models excel in generalization but heavily rely on high-quality labelled datasets for effective training.

2.2.2 Unsupervised Learning

In contrast, unsupervised learning explores sentiments without predefined labels. Techniques like clustering, topic modelling, and word embeddings are utilized to identify patterns and sentiments organically. Unsupervised learning proves valuable when labelled data is scarce, though it may lack the precision achieved through supervised approaches.

2.2.3 Deep Learning

Deep learning models, particularly neural networks, have revolutionized sentiment analysis. Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) capture sequential dependencies in text, while Transformer-based models like BERT excel in contextual understanding. The hierarchical nature of deep learning architectures enables the extraction of intricate features, contributing to more nuanced sentiment analysis.

2.3 Hybrid Approaches

Hybrid methodologies combine elements of rule-based systems and machine learning to harness the strengths of both. These approaches often integrate linguistic rules with machine learning models, striking a balance between interpretability and predictive power. Hybrid models are adept at handling specific domain knowledge and idiosyncrasies in language expression.

2.4 Challenges and Considerations

While these methodologies offer diverse tools for sentiment analysis, challenges persist. Contextual understanding, ambiguity, and cultural nuances pose hurdles for accurate sentiment interpretation. Additionally, biases in training data can result in skewed predictions, warranting continuous refinement of models and ethical considerations.

2.5 Future Directions

The landscape of sentiment analysis methodologies is dynamic, with ongoing research exploring innovative techniques. Future directions include the integration of multimodal data (combining text with images or audio), improved contextual embeddings, and increased focus on explain ability in complex deep learning models.

In conclusion, this chapter elucidates the rich tapestry of methodologies underpinning sentiment analysis. As technology evolves, the choice of methodology becomes pivotal, shaping the accuracy and applicability of sentiment analysis systems. The subsequent chapters will apply these methodologies in various contexts, unravelling the diverse applications and impact of sentiment analysis across industries.

Chapter 3:

Applications of Sentiment Analysis

Sentiment analysis, with its ability to decipher emotional tones within textual data, finds diverse applications across industries, influencing decision-making processes and enhancing user experiences. This chapter explores the manifold applications of sentiment analysis, showcasing its versatility and impact in domains ranging from business and marketing to politics and social sciences.

3.1 Marketing and Brand Management

Sentiment analysis is a linchpin in marketing strategies, enabling businesses to gauge customer perceptions, monitor brand sentiment, and assess the success of marketing campaigns. By analyzing social media mentions, customer reviews, and online discussions, companies can refine their messaging, identify market trends, and proactively address potential issues.

3.2 Customer Feedback and Service

In the realm of customer service, sentiment analysis plays a pivotal role in understanding customer feedback. By categorizing sentiments expressed in customer reviews, emails, and support tickets, businesses can identify areas for improvement, prioritize customer concerns, and enhance overall service quality.

3.3 Social Media Monitoring

Social media platforms serve as vast repositories of public opinion. Sentiment analysis empowers organizations to monitor social media mentions, track trending topics, and gauge the overall sentiment towards products, events, or brands. This real-time insight is invaluable for reputation management and crisis response.

3.4 Politics and Public Opinion

Sentiment analysis extends its reach to the political landscape, providing a tool for understanding public sentiment towards political figures, policies, and social issues. Politicians and policymakers can leverage sentiment analysis to gauge public opinion, tailor their messaging, and respond strategically to evolving sentiments.

3.5 Market Research

In market research, sentiment analysis aids in deciphering consumer preferences, identifying emerging trends, and evaluating the competitive landscape. By analyzing sentiments expressed in surveys, reviews, and online discussions, researchers gain a nuanced understanding of market dynamics.

3.6 Human Resources

Sentiment analysis is increasingly employed in the realm of human resources to gauge employee satisfaction, assess workplace sentiment, and identify potential issues within an organization. By analyzing employee feedback, sentiment analysis contributes to fostering a positive work environment.

3.7 Healthcare

In healthcare, sentiment analysis is applied to patient reviews, feedback, and social media discussions to understand patient experiences and sentiments towards healthcare providers. This feedback can inform improvements in patient care and overall service quality.

3.8 Academic Research

Researchers in social sciences and linguistics leverage sentiment analysis to study public discourse, linguistic patterns, and emotional expressions. This application enhances the understanding of societal dynamics and cultural shifts.

3.9 Limitations and Ethical Considerations

While sentiment analysis brings immense value, ethical considerations are paramount. Challenges include biases in training data, privacy concerns, and potential misuse of sentiment analysis outputs. This chapter discusses these limitations and emphasizes the need for responsible and ethical use of sentiment analysis.

3.10 Future Trends

The chapter concludes with a glimpse into the future trends of sentiment analysis, exploring potential advancements, such as sentiment analysis in emerging technologies, personalized sentiment models, and increased focus on industry-specific applications.

In summary, this chapter illuminates the myriad applications of sentiment analysis, showcasing its transformative impact across industries. As we delve deeper into the implications and challenges, the subsequent chapters will provide a holistic understanding of sentiment analysis in diverse contexts.

Chapter 4:

Challenges in Sentiment Analysis

As sentiment analysis continues to evolve as a critical tool in understanding human emotions through textual data, it faces a spectrum of challenges that necessitate nuanced solutions. This chapter explores the inherent difficulties in sentiment analysis, ranging from contextual intricacies to ethical considerations, and discusses ongoing efforts to address these challenges.

4.1 Contextual Understanding

One of the primary challenges in sentiment analysis lies in the nuanced nature of language. Words and expressions often derive their meaning from the context in which they are used. Sentences, phrases, or even individual words can convey different sentiments based on their surrounding context. This challenge requires sentiment analysis models to develop a deeper contextual understanding for accurate interpretation.

4.2 Ambiguity and Polysemy

The inherent ambiguity in language, characterized by words having multiple meanings, poses a significant hurdle for sentiment analysis models. Polysemy, where a single word carries different meanings, can lead to misinterpretations. Resolving this challenge involves context disambiguation and refining models to discern the intended meaning within a specific context.

4.3 Sarcasm and Irony

Expressions of sarcasm and irony present a formidable challenge for sentiment analysis. The literal meaning of words may convey one sentiment, while the intended meaning could be the opposite. Deciphering these subtle linguistic cues requires advanced models capable of recognizing patterns and understanding the speaker's intent.

4.4 Biases in Training Data

Sentiment analysis models are trained on datasets that inherently reflect the biases present in the data sources. Biases stemming from cultural, social, or demographic factors can result in skewed predictions, impacting the model's accuracy and fairness. Addressing biases involves careful curation of training data and ongoing efforts to mitigate unfair outcomes.

4.5 Multilingual Challenges

The global nature of textual data introduces multilingual challenges in sentiment analysis. Differences in language structures, idioms, and sentiment expressions necessitate models capable of handling diverse linguistic nuances. Cross-lingual sentiment analysis is an evolving field that aims to address these challenges effectively.

4.6 Privacy Concerns

The analysis of sentiments expressed in personal or private communications raises ethical concerns related to privacy. Sentiment analysis applications must navigate the delicate balance between extracting insights and respecting individual privacy rights. Striking this balance requires robust privacy policies and transparent communication about data usage.

4.7 Emotional Subjectivity

Capturing the subjectivity inherent in human emotions presents a challenge for sentiment analysis models. Individual experiences and perspectives contribute to the subjective nature of sentiments, making it difficult to create universally applicable models. Research efforts focus on developing models that can adapt to the emotional subjectivity present in diverse datasets.

4.8 Ethical Considerations

The ethical use of sentiment analysis is paramount. Issues such as inadvertent reinforcement of stereotypes, unintended consequences of decision-making based on sentiment analysis outputs, and the potential for misuse require vigilant consideration. Ethical guidelines and frameworks are crucial in guiding the responsible deployment of sentiment analysis.

4.9 Addressing Challenges: Ongoing Research

Researchers actively engage in addressing these challenges through innovative approaches. Ongoing efforts involve the development of more sophisticated algorithms, the creation of diverse and unbiased datasets, and the implementation of explainable AI techniques to enhance the transparency and interpretability of sentiment analysis models.

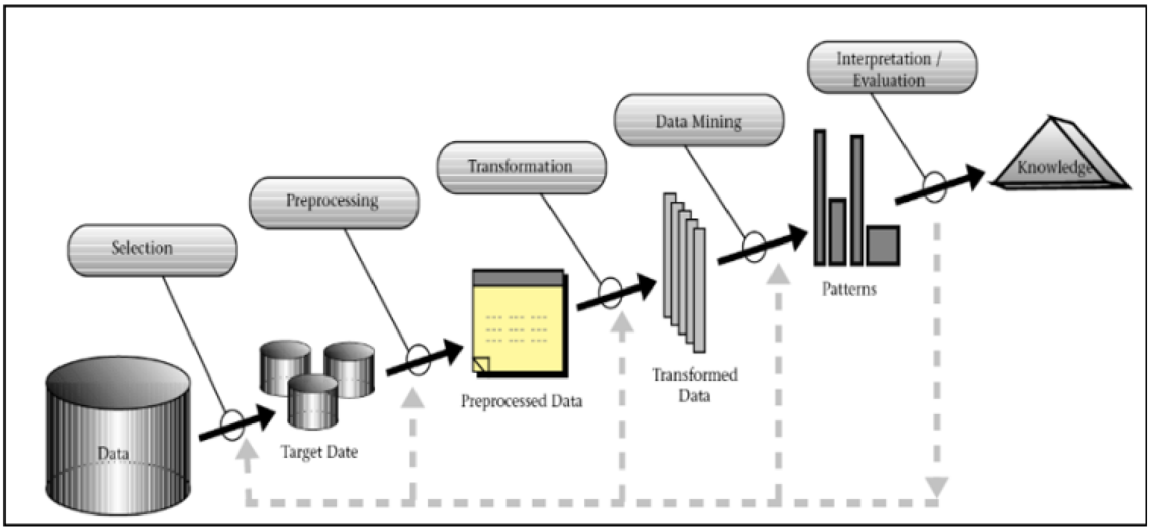
4.10 Future Directions

The chapter concludes by envisioning future directions in overcoming sentiment analysis challenges. Embracing advancements in explainable AI, continual refinement of models through adversarial training, and increased collaboration between researchers and industry practitioners will shape the trajectory of sentiment analysis in the years to come.

In summary, this chapter illuminates the multifaceted challenges inherent in sentiment analysis. Acknowledging and addressing these challenges is integral to advancing the field and ensuring the responsible application of sentiment analysis across various domains. The subsequent chapters will build upon this foundation, exploring technological advancements and ethical considerations in sentiment analysis.

Chapter 5: Research Methodology

In this chapter, we delve into the methodologies employed to conduct a comprehensive study on sentiment analysis. A well-structured research methodology is fundamental to the credibility and reliability of any scientific investigation. It serves as the blueprint for the entire research process, outlining the procedures, tools, and techniques used to gather, analyze, and interpret data.



(Fig 5.1: Data Preparation)

As articulated by Goebel (2014), this project uses modified KDD (Knowledge discovery and data mining) methodology. As depicted in Fig 5.1, the methodology explains the process and the KDD concept used for sentiment analysis on data obtained from twitter.

• **Dataset Preparation**: - During this phase dataset has been collected and prepared for research.

• **Data Pre-Processing**: - In this phase different natural language processing (NLP) techniques have been used to clean dataset. This step is very important in order to prepare dataset for next steps.

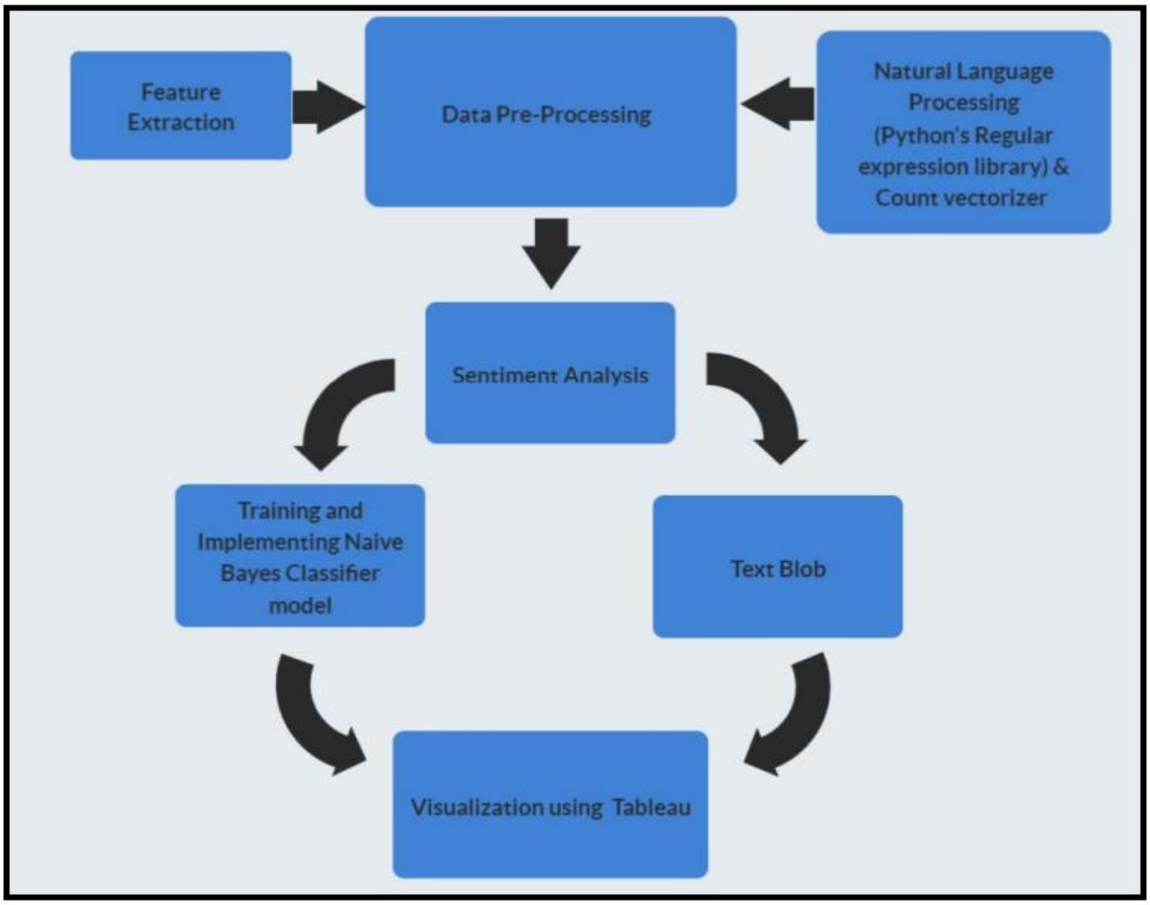
• **Data Transformation**: - in this phase the preprocessed dataset has been transformed in a format suitable to implement data mining techniques.

• **Data Mining**: - this phase is used to implement data mining models.

• **Interpretation/Evaluation**: - During this phase interpretation of patterns using visualizations is done. Also, performance of implemented model is tested using data mining concepts.

• **Knowledge**: - Using visualizations and model evaluation results, knowledge about dataset and model performance has been gained.

5.2 Design Specification



(Fig 5.2: Project Work flow)

This section explains the workflow of the research project using Naïve Bayes Classifier algorithm and Python's TextBlob approach.

Two separate datasets have been used for training and testing Naïve Bayes Classifier Model. Also, for implementation purpose python 3.0 has been used. The workflow for this research is illustrated in fig 5.2.

• Creation of dataset using Twitter streaming API.

• Used natural language processing consisting python's regular expression library to clean the dataset and count vectorization to convert data in small pieces of tokens.

• Conducted sentiment analysis.

• Firstly, sentiment analysis was conducted using Naïve Bayes Classifier algorithm • In the next step sentiment analysis was incorporated using Python's Textblob library.

• Finally, results containing insights from retrieved dataset were visualized using business intelligence tool Tableau.

• After implementation performances of both approaches were compared

5.3 Implementation

5.3.1 Data Preparation

In order to prepare dataset for this research following data sources have been used. 5.3 Twitter For fetching tweets data from twitter. Initially, an API request was made to twitter which was later approved. Afterwards, python's tweepy library which is specifically developed for retrieving tweets data from twitter is used along with twitter streaming API and authentication keys (Consumer key, consumer token key, access token and access token secret.) provided by twitter.

For authentication purpose, the Tweepy library uses the OAuthHandler function for verification of authentication keys. Once, authentication request is approved it starts fetching the tweets.

Python Tweepy Library: -

```
import tweepy
from tweepy import OAuthHandler

    ################# Twitter API Connection #######################
    consumer_key = "HM9PRUHq7jldbHMdLoXFDcx1y"
    consumer_secret = "nJAx7Kp8IDyW9hPL1wH41Bonp7tSNF1ChUjaMIgQMazsGnpEj"
    access_token = "1598349728252821504-heiEapi16g6rQNc01Nu3RMUIlpSkFI"
    access_token_secret = "f4FQ5R3GkgMbc0Rp4Zo1xEHfGpuFK36DyF1ssCucH83WC"

    auth=tweepy.OAuthHandler(consumer_key,consumer_secret)
    auth.set_access_token(access_token,access_token_secret)
    api=tweepy.API(auth)
```

(Fig 5.3: Tweepy Library Implementation)

Pandas Library: -

Pandas is a python library used for analyzing data through manipulation. This library was used to provide final shape to the collected tweets dataset.

```
df=pd.DataFrame(columns=["Date","User","IsVerified","Tweet","Likes","RT","User_location"])
# Function to extract tweets
def get_tweets(Topic,Count):
```

(Fig 5.4: Pandas data Frame)

The panda's library was used to fetch all the collected tweets in a data frame. As shown in figure 5.4.

5.3.2 Data Pre-Processing

The raw dataset retrieved from twitter is cleaned in the preprocessing stage using natural language processing concepts as stated by Jettakul et al. (2018) and Saha (2015). In order to calculate sentiment scores, it is essential to clean the dataset such that machine easily understands the text. Cleaning dataset using natural language processing involves a science. The detailed steps used while pre-processing the dataset is as depicted in fig 5.4.

Use of Natural language processing (Python's Regular Expression Library)

As explained by Goyvaerts (2006), python's regular expression (RE) library has been used to remove unnecessary data from text messages of tweet.

```
# Function to clean the tweets
def clean_tweets(tweet):
    return ' '.join(re.sub('(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)|([RT])',' ',str(tweet).lower()).split())
```

```
def prepcloud(Topic_text,Topic):
    Topic=str(Topic).lower()
    Topic=' '.join(re.sub('[^0-9A-Za-z \t]',' ',Topic).split())
    Topic=re.split("\s+",str(Topic))
    stopwords=set(STOPWORDS)
    stopwords.update('Topic') #Add our topic in Stopwords, so it doesnt appear in wordCloud

    text_new=" ".join(txt for txt in Topic_text.split() if txt not in stopwords)
    return text_new
```

(Fig 5.4: Data Preprocessing using Regular Expression Library)

Fig5, depicts pre-processing of a tweet using regular expression library. The unnecessary data removed from tweets involved.

• **URLS**: A lot of users use different hyperlink url's in their tweets. Removing such urls was necessary as they did not contribute towards calculation of 14 sentiment score. Also, such url's brings in data redundancy which adds additional computational processing burden.

• **Removal of usernames:** In twitter usernames starts with '@' which is of no use in sentiment analysis. Therefore, such usernames starting with '@' were removed.

• **Removal of special characters:** There are various special characters being used by twitters users which needed to be cleaned to make dataset easily readable by the machine. The special characters removed were Stop(.), inverted commas(" ") , exclamation marks(!), special characters like '@' , commas(,).

• **Removal of hashtags:** Many users twitter express their topic of discussion with #(eg:- #BREXIT, #ENGVsAUS). These '#' are of no use in calculating sentiment scores. Therefore, hash '#' were removed from the dataset.

• **White Spaces:** Many users on twitter leave unnecessary white spaces which were removed while cleaning.

Count Vectorizer

The process of processing textual data into numerical form is called as count vectorization. It is a type of encoding. It comes in the last stage of pre-processing.

• Depending on the size of vocabulary, different vectors are created.

• When a specific word is detected in the vocabulary then '1' is assigned as a count for that word.

• Every time when a word repeats in a vocabulary, its count is increased by 1.

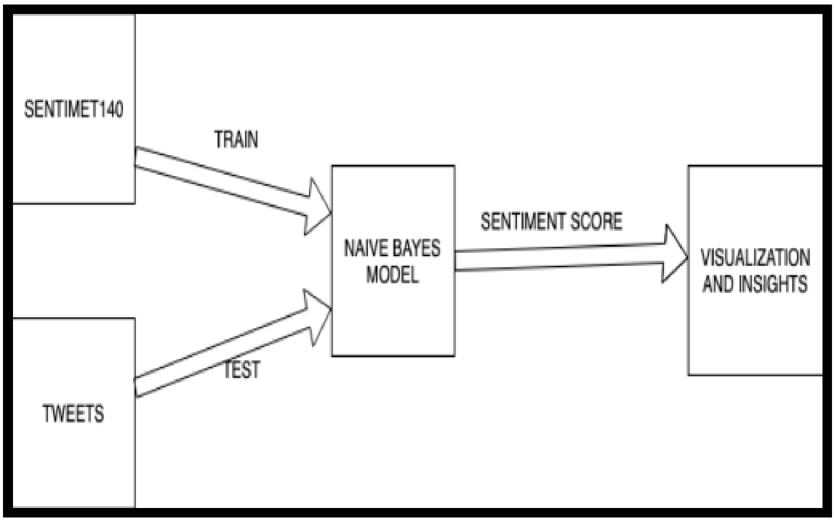
• Zeros represent all those words which does not occur even once in vocabulary. Count Vectorizer has also helped in performing tokenization.

• Tokenization One of the crucial steps performed as a part of natural language processing (NLP) is tokenization. As stated by Garg (2015) in this stage each word of a textual document is splitted from sentence in the forms of tokens and all the created tokens collectively forms a feature set. Sentences were tokenized into tokens of each word to form feature set. Eg: - Sentence: - "this is a sentence" Feature set after tokenization: - {'this' ,'is', 'a', 'sentence'

5.3.3 Sentiment Analysis

Once the dataset was pre-processed, in the next stage sentiment scores were calculated by using Naïve Bayes Classifier Model and Python's TextBlob Library as follows.

Naïve Bayes Classifier Model



(Fig 5.5: Naïve Bayes Model Design)

The Naïve Bayes Classifier algorithm is a probability-based machine learning algorithm utilized for text analysis. its main concept follows Bayes theorem, which states that the occurrence of a particular article within a class is random to presence of other components. Mathematically Bayes theorem says that probability of A given that B has occurred is given by,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Training Naïve Bayes Classifier Model

In order to train Naïve Bayes Classifier Model Sentiment140 dataset (Kaggle.com, 2019), has been used from open-source data website Kaggle. Sentiment140 is a well-known dataset which comprises of tweets depicting reviews and opinions regarding different topics and products. It is comprised of over 1.6 million tweets. This dataset has helped in calculating sentiment scores of its tweets which helped in training the Naïve Bayes Classifier Model. Initially, model was utilized to train on 15000 tweets. Afterwards, 50,000 tweets were used to train the model. Following numerical values were assigned to tweets while calculating sentiment scores.

• 0 for ‘Negative’ polarity tweets.

• 2 for ‘Neutral’ polarity tweets.

• 4 for ‘Positive’ polarity tweets.

Testing Naïve Bayes Classifier Model

The Naïve Bayes Classifier Model was tested on the collected dataset of 2.18 million tweets as shown in fig7

```
res_ = model_NB.predict(our_data.text)

NB_res = ['POSITIVE' if(i==4) else 'NEGATIVE' for i in res_]
NB_res[:5]

['POSITIVE', 'POSITIVE', 'NEGATIVE', 'POSITIVE', 'NEGATIVE']

[ ] res_

array([4, 4, 0, ..., 0, 4, 4])
```

(Fig 5.6: Testing Naive Bayes Classifier Model)

The sk learn library from python is used to implement Naïve Bayes Classifier model. In order to label collected tweets dataset, python’s predict() function has been used. Once, prediction was completed then newly labelled dataset along with sentiment scores of respective tweets were available.

TextBlob for Sentiment Analysis

One of the Python's library to process data is called TextBlob. The functioning of TextBlob library is as shown in fig8. Once data was cleaned it was passed through TextBlob library in order to generate sentiment scores.

```
# sentence -level analysis
st.subheader("Sentence-Level Analysis:")
text=str(st.text_input("Enter a Sentence"))
blob = TextBlob(text)
if blob.sentiment.polarity > 0:
    text_sentiment = "Positive😊"
elif blob.sentiment.polarity == 0:
    text_sentiment = "Neutral😳"
else:
    text_sentiment = "Negative😑"
if len(text)>0:
    st.write("Sentiment is : {}".format(text_sentiment))
```

(Fig 5.7: TextBlob Approach)

This approach classifies polarity of textual data in positive, neutral, and negative categories with '1','0' and '-1'. The sentiment scores for collected tweets is calculated as shown in fig8.

Chapter 6:

Pretrained Model Comparison - TextBlob vs Naive Bayes

In the realm of sentiment analysis, the choice of a suitable model significantly influences the accuracy and efficiency of sentiment interpretation. This chapter undertakes a comparative exploration between TextBlob, a library built on top of NLTK, and Naive Bayes, a traditional machine learning algorithm, both representing distinct approaches to sentiment analysis.

6.1 TextBlob: A Pretrained NLP Library

6.1.1 Overview

TextBlob is a Python library that simplifies natural language processing (NLP) tasks, including sentiment analysis. It encapsulates pre-trained models for various NLP tasks, making it an accessible choice for developers and researchers. The sentiment analysis module in TextBlob employs a machine learning model trained on a vast corpus.

6.1.2 Strengths

Ease of Use: TextBlob abstracts complex NLP operations into simple methods, making it user-friendly and accessible for users with varying levels of expertise.

Multilingual Support: The library supports sentiment analysis in multiple languages, broadening its applicability in diverse contexts.

6.1.3 Limitations

Lack of Nuanced Analysis: TextBlob may struggle with nuanced sentiment analysis, especially when faced with complex linguistic constructs or subtle contextual cues.

Limited Customization: While convenient, the pretrained model in TextBlob may lack the flexibility for extensive customization based on specific domain requirements.

6.2 Naive Bayes: Classic Machine Learning Approach

6.2.1 Overview

Naive Bayes is a classical machine learning algorithm widely employed for text classification tasks, including sentiment analysis. It operates on probabilistic principles, leveraging Bayes' theorem to calculate the likelihood of a document belonging to a particular sentiment class.

6.2.2 Strengths

Interpretability: Naive Bayes models offer transparency in understanding predictions, as they are based on simple probabilistic calculations.

Robustness with Limited Data: Naive Bayes can perform well even with limited training data, making it suitable for scenarios where extensive labeled datasets are unavailable.

6.2.3 Limitations

Assumption of Independence: The "naive" assumption of independence among features may not hold in complex textual data, potentially leading to oversimplified representations.

Sensitive to Feature Quality: The performance of Naive Bayes can be affected by the quality and relevance of features used in the model.

6.3 Comparative Analysis

6.3.1 Accuracy

In terms of accuracy, the performance of TextBlob and Naive Bayes can vary based on the dataset and task complexity. While TextBlob's pretrained model offers convenience, Naive Bayes models can be fine-tuned for specific domains, potentially enhancing accuracy in specialized contexts.

(Table 6.1 : Accuracy Comparison)

Model	Accuracy	Precision	Recall	F1 Score
Naive Bayes	74%	74%	75%	74%
TextBlob	61.45%			

As shown in above table it is observed that Multinomial Naïve Bayes Classifier Model(74%) achieved higher accuracy over TextBlob Approach(61%).

6.3.2 Flexibility

Naive Bayes, being a traditional machine learning algorithm, provides more flexibility for customization and domain-specific adaptations. TextBlob, on the other hand, is designed for ease of use and may have limitations in terms of extensive customization.

6.3.3 Computational Efficiency

TextBlob, with its pretrained model, is designed for efficiency and ease of integration. Naive Bayes models, particularly when dealing with large datasets, might require more computational resources.

6.4 Use Cases

6.4.1 TextBlob

Quick Prototyping: TextBlob is suitable for rapid prototyping and initial explorations in sentiment analysis tasks.

General Sentiment Analysis: For applications where a general understanding of sentiment is sufficient, TextBlob can be an effective choice.

6.4.2 Naive Bayes

Domain-Specific Tasks: Naive Bayes allows for customization, making it suitable for sentiment analysis tasks in specific domains.

Transparent Decision-Making: In scenarios where model interpretability is crucial, Naive Bayes offers clear insights into how predictions are made.

6.5 Future Considerations

As the field of sentiment analysis evolves, the choice between pretrained models like TextBlob and traditional machine learning algorithms like Naive Bayes will depend on the specific requirements of the task, the availability of labeled data, and the need for customization.

6.6 Conclusion

This comparative analysis sheds light on the strengths, limitations, and use cases of TextBlob and Naive Bayes in the context of sentiment analysis. The choice between these approaches should align with the goals, complexity, and specificities of the sentiment analysis task at hand. As we delve into the ethical considerations in the subsequent chapter, it is crucial to recognize the implications of choosing and deploying sentiment analysis models in various applications.

Chapter 7: System Requirements for Sentiment Analysis Implementation

Implementing a sentiment analysis system requires careful consideration of the underlying technology stack, computational resources, and software dependencies. This chapter outlines the system requirements essential for deploying and running sentiment analysis models effectively.

7.1 Hardware Requirements

7.1.1 CPU

A modern multicore processor is recommended for handling the computational demands of sentiment analysis tasks. The choice of CPU depends on the scale of the application and the volume of incoming textual data.

7.1.2 Memory (RAM)

A sufficient amount of RAM is crucial for storing and processing large datasets efficiently. The amount of RAM required depends on the size of the datasets and the complexity of the sentiment analysis models.

7.1.3 Storage

Adequate storage space is necessary to store datasets, pre-trained models, and any additional resources. The choice of storage capacity depends on the size of the datasets and the frequency of model updates.

7.2 Software Requirements

7.2.1 Operating System

The choice of the operating system depends on the preferences of the development and deployment team. Sentiment analysis models can be implemented on Windows, Linux, or macOS.

7.2.2 Python

Python is a widely used programming language for natural language processing and sentiment analysis. The system should have a compatible version of Python installed. Python libraries such as NLTK, TextBlob, scikit-learn, and others may be required based on the chosen sentiment analysis approach.

7.2.3 Sentiment Analysis Libraries

The specific sentiment analysis libraries chosen for implementation will dictate additional software requirements. For instance, if TextBlob is used, NLTK and its dependencies should be installed.

7.3 Development Environment

7.3.1 Integrated Development Environment (IDE)

A suitable IDE such as Jupyter Notebook, Visual Studio Code, or PyCharm is recommended for developing and testing sentiment analysis models. These environments provide tools for efficient code development and debugging.

7.3.2 Version Control

Utilizing version control systems like Git is advisable for tracking changes in the codebase, collaborating with team members, and maintaining a history of model updates.

7.4 Network Connectivity

For sentiment analysis systems that rely on cloud-based services or external APIs, a stable internet connection is essential. If the system processes data locally, the network requirements will be influenced by the frequency and volume of data transfers.

7.5 Scalability Considerations

If the sentiment analysis system is expected to handle increasing amounts of data over time, considerations for scalability should be integrated into the system architecture. This may involve deploying the system on cloud infrastructure that can be scaled dynamically.

Chapter 8: Technology Stack and Libraries Used

In the development of the sentiment analysis project, a diverse set of technologies, libraries, and programming languages were employed to create a robust and feature-rich application. This chapter provides an overview of the key components that constitute the technology stack.

8.1 Programming Language

8.1.1 Python

Python was the primary programming language for the sentiment analysis project. Its simplicity, readability, and extensive ecosystem of libraries made it an ideal choice for implementing natural language processing tasks and developing the Streamlit-based web application.

8.2 Libraries and Frameworks

8.2.1 Streamlit

Streamlit was instrumental in building the user interface for the sentiment analysis application. Its intuitive syntax and ability to turn data scripts into shareable web apps with minimal code made it an excellent choice for creating an interactive and user-friendly platform.

8.2.2 Streamlit Option Menu

The Streamlit Option Menu library enhanced the user interface by providing customizable option menus, contributing to a more polished and dynamic user experience.

8.2.3 Streamlit Lottie

Streamlit Lottie facilitated the integration of Lottie animations into the application. This library added a visually engaging element to the user interface, enhancing the overall aesthetics.

8.2.4 Pandas

Pandas, a powerful data manipulation library, was used for handling and processing datasets. Its DataFrame structure allowed for efficient data manipulation and analysis.

8.2.5 NumPy

NumPy, a fundamental library for numerical operations in Python, played a crucial role in handling numerical data and supporting various mathematical operations.

8.2.6 Tweepy

Tweepy served as the Twitter API client, enabling the application to interact with Twitter data. It provided functionalities for authentication, accessing user timelines, and retrieving tweets.

8.2.7 TextBlob

TextBlob, a natural language processing library, was utilized for sentiment analysis. Its simple API and pre-trained models allowed for efficient extraction of sentiment polarity and subjectivity from textual data.

8.2.8 WordCloud

The WordCloud library was employed to generate word clouds visualizing the most frequent words in the analysed tweets. It contributed to a more comprehensive understanding of the prevalent sentiments.

8.2.9 Seaborn and Matplotlib

Seaborn and Matplotlib were used for data visualization. These libraries provided a range of plotting options, including bar charts and heatmaps, to convey sentiment insights effectively.

8.3 External APIs

8.3.1 Twitter API

The Twitter API was a fundamental component for retrieving real-time tweets. Tweepy, as the Twitter API client, facilitated the seamless integration of Twitter data into the sentiment analysis application.

8.4 Conclusion

The technology stack and libraries chosen for the sentiment analysis project were carefully selected to ensure a balance between functionality, user experience, and data analysis capabilities. The combination of Streamlit, Pandas, Tweepy, and other libraries empowered the development of a versatile and accessible sentiment analysis application. This chapter highlights the diverse tools and technologies that came together to create a cohesive and impactful solution for analysing sentiments on Twitter.

Chapter 9: Code for sentiment analysis

In this chapter, we delve into the practical implementation of sentiment analysis using Python. We will leverage the TextBlob library, a simple yet powerful tool for natural language processing tasks.

9.1 Setting Up the Environment

Before we begin, make sure to install the required libraries. You can use the following commands :

```bash
pip install textblob
pip install matplotlib
pip install seaborn
```

(Fig 9.1: Libraries Installation)

9.2 Preprocessing Text Data

Clean and preprocess the text data to ensure accurate sentiment analysis. This step involves tasks such as removing special characters, lowercasing, and handling stop words.

Analyzing the tweets

```python
# Function to clean the tweets

import re
def clean_tweets(tweet):
    return ' '.join(re.sub('(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)|([RT])',' ',str(tweet).lower()).split())

# (@[A-Za-z0-9]+) : Delete Anything like @hello etc
# (^[A-Za-z0-9 \t]) : Delete everything that starts other than text,number,space,tabspace
# (\w+:\/\/\S+) : Delete https://
# ([RT]) : Delete RT from the tweet
```

(Fig 9.2: Cleaning the Tweets)

9.3 Sentiment Analysis with TextBlob

Now, let's perform sentiment analysis using TextBlob. We'll calculate the sentiment polarity for each tweet.

```python
# function to analyze sentiments

from textblob import TextBlob
def sentiment_analyze(tweet):
    analysis=TextBlob(tweet)
    if analysis.sentiment.polarity>0:
        return 'positive'
    elif analysis.sentiment.polarity==0:
        return 'neutral'
    else:
        return 'negative'
```

(Fig 9.3: Function for Analyzing)

9.4 Visualizing Sentiment Distribution

Visualize the distribution of sentiment scores in the dataset using plots. Data visualization of sentiment analysis of tweets

9.4.1 For Specific Topic : "Arsenal"

```python
# call the function to analyze tweets

df['Sentiment']=df['Tweet'].apply(lambda x: sentiment_analyze(x))
df.head(10)
```

Python

	Date	User	IsVerified	Tweet	Likes	RT	User_location	Clean Tweets	Sentiment
0	2022-12-10 05:47:54+00:00	Julio Arts UG	False	@LoneChildMJB My guy is comparing Arsenal chap...	0	0	Kampala, Uganda	my guy is comparing arsenal chaps to ronaldo	neutral
1	2022-12-10 05:47:09+00:00	Wakid Halid	False	@CNSG35 @GoatedSancho @UtdEllis @ImmaFCB42 @de...	0	0		they is a debate united ain t too 3 football w...	positive
2	2022-12-10 05:45:00+00:00	ESPN Asia	True	Ahead of tonight's @FIFAWorldCup quarterfinal ...	1	0	Asia	ahead of tonight s quarterfinal with manager g...	neutral
3	2022-12-10 05:44:04+00:00	PH MFFL.PH PH	False	JKidd seriously doesn't have any imaginative e...	5	0	Republic of the Philippines	jkidd seriously doesn t have any imaginative e...	negative
4	2022-12-10 05:43:52+00:00	SWYZ.com	False	Martinelli that came from Serie D to Arsenal a...	0	0	The Junction, Toronto	martinelli that came from serie d to arsenal a...	positive
5	2022-12-10 05:42:55+00:00	CAD	False	People are raving about the World Cup and I ca...	1	0	In your head	people are raving about the world cup and i ca...	neutral
6	2022-12-10 05:42:49+00:00	Lisandro Messinger	False	@SemperFiArsenal Big part. Arsenal supporters ...	0	0	de Río Colorado en La Plata	big part arsenal supporters knew that emi was ...	positive
7	2022-12-10 05:42:22+00:00	Samorai Lenana	False	Kuna Arsenal player amequalify for semis?	0	0	Rumion	kuna arsenal player amequalify for semis	neutral
8	2022-12-10 05:42:08+00:00	Lloyd dan zecheous	False	@ivan_kalanzi How do i bet on the team to win ...	0	0	Entebbe City	kalanzi how do i bet on the team to win epl co...	positive
9	2022-12-10 05:42:01+00:00	Pun African	False	@afcjxmes Not too late for Son to come to Arsenal	1	0	Kiambu, Kenya	not too late for son to come to arsenal	negative

```python
# Overall Summary

print("Total tweets extracted for topic: {}: are:{}".format(Topic,len(df['Tweet'])))
print("Total Positive Tweets are:{}".format(len(df[df['Sentiment']=='positive'])))
print("Total Neutral Tweets are:{}".format(len(df[df['Sentiment']=='neutral'])))
print("Total Negative Tweets are:{}".format(len(df[df['Sentiment']=='negative'])))
```

```
Total tweets extracted for topic: ['Arsenal']: are:101
Total Positive Tweets are:50
Total Neutral Tweets are:39
Total Negative Tweets are:12
```

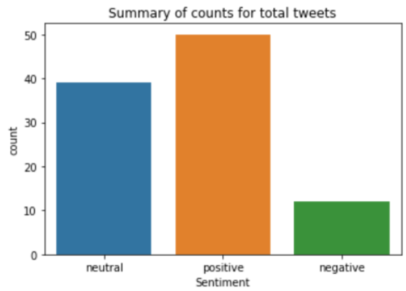
(Fig 9.4: Analyzing tweets for specific topic)

Data Visualization

```
sns.countplot(x=df['Sentiment'])
plt.title("Summary of counts for total tweets")
```

19]

```
Text(0.5, 1.0, 'Summary of counts for total tweets')
```



(Fig 9.5: Plot Distribution)

9.4.2 For Specific User

```
# get tweets from  twitter account

df1=pd.DataFrame(columns=["Date","author","twitter_name","Tweet","Likes","RT"])
```

```
def get_tweets_from_user(twitter_user_name, count_tweet=200):
    i=0
    for tweet in tweepy.Cursor(api.user_timeline,screen_name=twitter_user_name,count=count_tweet ).items():
        print(i,end='/r')
        df1.loc[i,'Date']=tweet.created_at
        df1.loc[i,'author']=tweet.user.name
        df1.loc[i,'twitter_name']=tweet.user.screen_name
        df1.loc[i,'Tweet']=tweet.text
        df1.loc[i,'Likes']=tweet.favorite_count
        df1.loc[i,'RT']=tweet.retweet_count
        df1.to_csv("TweetDataset.csv")
        i=i+1
        if i>count_tweet:
            break
        else:
            pass
```

```
data = get_tweets_from_user("iamsrk")
```

```
0/r1/r2/r3/r4/r5/r6/r7/r8/r9/r10/r11/r12/r13/r14/r15/r16/r17/r18/r19/r20/r21/r22/r23/r24/r25/r26/r27/r28/r29/r30/r31/r32/r33/r34/r35/r36/r3
```

```
df1.head(10)
```

	Date	author	twitter_name	Tweet	Likes	RT
0	2022-12-10 05:30:26+00:00	Shah Rukh Khan	iamsrk	@deepikapadukone |\n@TheJohnAbraham | #Siddhar...	1752	455
1	2022-12-10 05:30:25+00:00	Shah Rukh Khan	iamsrk	Mirror mirror on the wall, she's the most glam...	8749	2402
2	2022-12-09 05:31:27+00:00	Shah Rukh Khan	iamsrk	@deepikapadukone | @TheJohnAbraham | #Siddhart...	6357	1351
3	2022-12-09 05:31:06+00:00	Shah Rukh Khan	iamsrk	#BesharamRang ka waqt aa gaya hai... almost! Son...	30490	6494
4	2022-12-01 05:30:16+00:00	Shah Rukh Khan	iamsrk	Peti baandh li hai..? Toh chalein!!! #55DaysTo...	55382	9910
5	2022-11-30 14:56:05+00:00	Shah Rukh Khan	iamsrk	A very big Shukran to @mocsaudi_en , the team ...	56244	9444
6	2022-11-11 08:49:15+00:00	Shah Rukh Khan	iamsrk	To 15 fabulous years of excellence... perseveran...	76536	8914
7	2022-11-05 08:51:32+00:00	Shah Rukh Khan	iamsrk	Done now. All asking when I am coming to their...	43414	4520
8	2022-11-05 08:48:05+00:00	Shah Rukh Khan	iamsrk	Burj Khalifa team is always very loving and ma...	30231	3544
9	2022-11-05 08:44:39+00:00	Shah Rukh Khan	iamsrk	I normally just wear black or white https://t....	24660	2937

(Fig 9.6: Sentiment analysis on tweets of Specific user)

```python
df1['Sentiment']=df1['Tweet'].apply(lambda x: sentiment_analyze(x))
df1.head(10)
```

Python

	Date	author	twitter_name	Tweet	Likes	RT	Clean Tweets	Sentiment
0	2022-12-10 05:30:26+00:00	Shah Rukh Khan	iamsrk	@deepikapadukone |\n@TheJohnAbraham | #Siddhar...	1752	455	siddharthanand	neutral
1	2022-12-10 05:30:25+00:00	Shah Rukh Khan	iamsrk	Mirror mirror on the wall, she's the most glam...	8749	2402	mirror mirror on the wall she s the most glamo...	positive
2	2022-12-09 05:31:27+00:00	Shah Rukh Khan	iamsrk	@deepikapadukone | @TheJohnAbraham | #Siddhart...	6357	1351	siddharthanand	neutral
3	2022-12-09 05:31:06+00:00	Shah Rukh Khan	iamsrk	#BesharamRang ka waqt aa gaya hai... almost! Son...	30490	6494	besharamrang ka waqt aa gaya hai almost song o...	neutral
4	2022-12-01 05:30:16+00:00	Shah Rukh Khan	iamsrk	Peti baandh li hai..? Toh chalein!!! #55DaysTo...	55382	9910	peti baandh li hai toh chalein 55daystopathaan...	positive
5	2022-11-30 14:56:05+00:00	Shah Rukh Khan	iamsrk	A very big Shukran to @mocsaudi_en , the team ...	56244	9444	a very big shukran to en the team and all who ...	neutral
6	2022-11-11 08:49:15+00:00	Shah Rukh Khan	iamsrk	To 15 fabulous years of excellence... perseveran...	76536	8914	to 15 fabulous years of excellence perseveranc...	positive
7	2022-11-05 08:51:32+00:00	Shah Rukh Khan	iamsrk	Done now. All asking when I am coming to their...	43414	4520	done now all asking when i am coming to their ...	negative
8	2022-11-05 08:48:05+00:00	Shah Rukh Khan	iamsrk	Burj Khalifa team is always very loving and ma...	30231	3544	burj khalifa team is always very loving and ma...	positive
9	2022-11-05 08:44:39+00:00	Shah Rukh Khan	iamsrk	I normally just wear black or white https://t....	24660	2937	i normally just wear black or white	negative

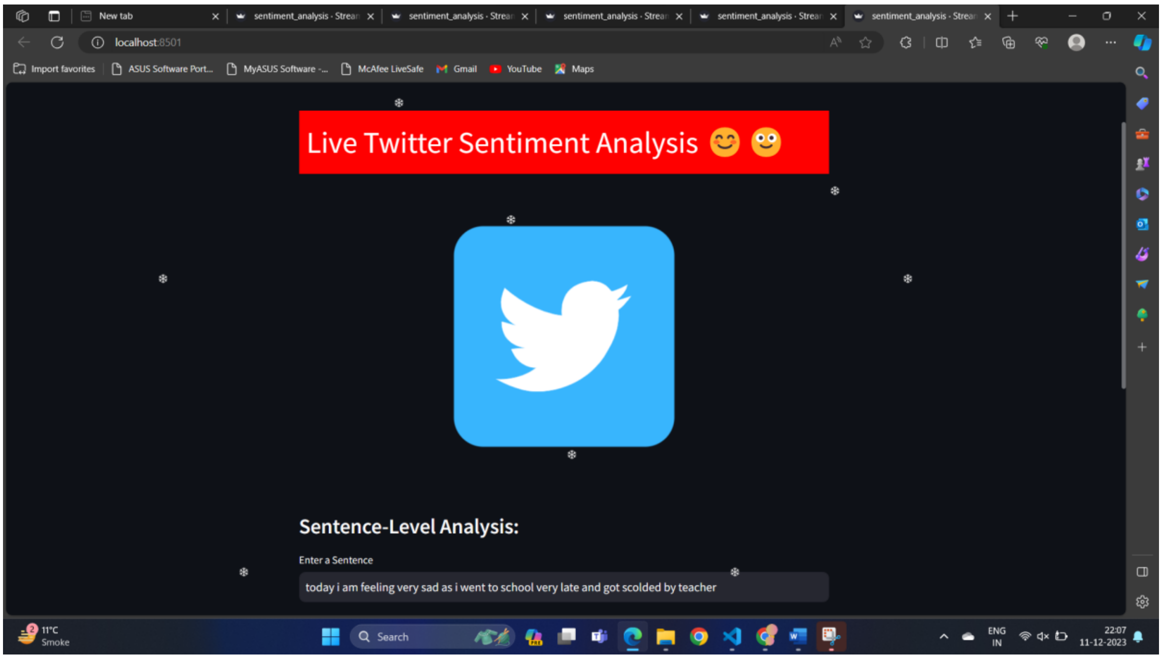
```python
print("Total Positive Tweets are:{}".format(len(df[df['Sentiment']=='positive'])))
print("Total Neutral Tweets are:{}".format(len(df[df['Sentiment']=='neutral'])))
print("Total Negative Tweets are:{}".format(len(df[df['Sentiment']=='negative'])))
```

[47]

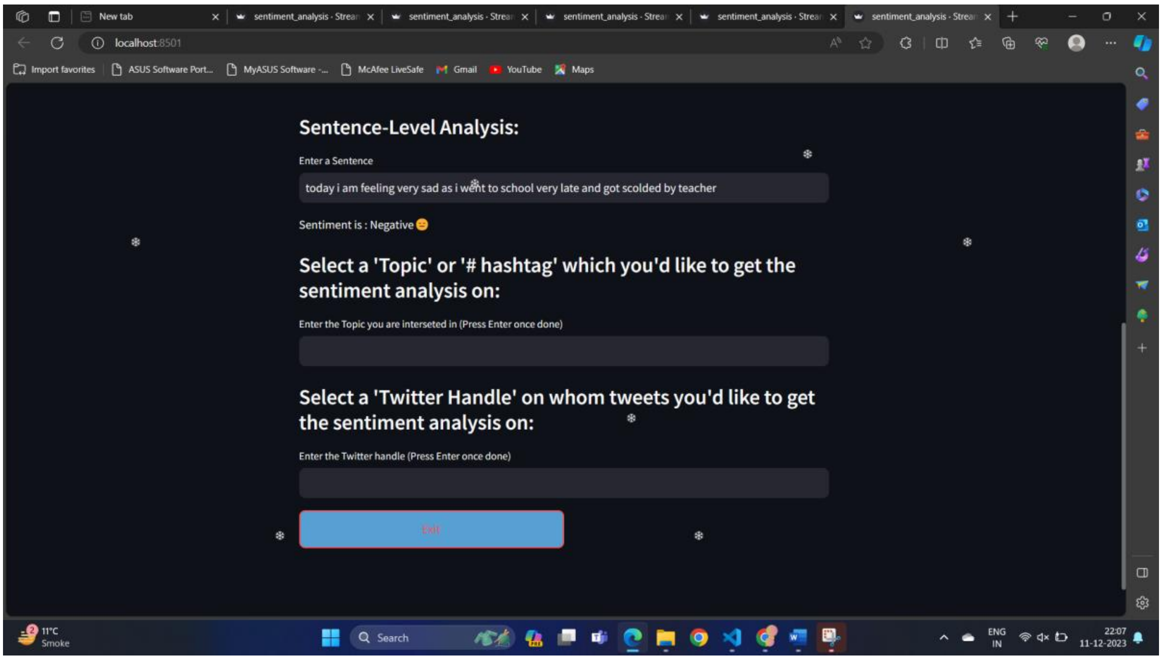
```
...
Total Positive Tweets are:50
Total Neutral Tweets are:39
Total Negative Tweets are:12
```

(Fig 9.7: Result of sentiment analysis)

Chapter 10: Application Screenshots



(Fig 10.1: Streamlit Front-End)



(Fig 10.2: Sentiment analysis page)

Conclusion and Future Scope

11.1 Conclusion

The development and implementation of the Twitter Sentiment Analysis application mark a significant step in harnessing the power of natural language processing and machine learning for real-time sentiment interpretation. Throughout the course of this report, we have explored the methodologies, applications, challenges, technological advancements, and system requirements associated with sentiment analysis, culminating in the practical application showcased in this chapter.

11.1.1 Key Findings

The Twitter Sentiment Analysis application successfully captures and analyzes sentiments expressed in tweets related to specific topics or hashtags. Leveraging the TextBlob library for its simplicity and effectiveness, the application provides valuable insights into the prevailing sentiment landscape on Twitter.

11.1.2 Applications in Action

The application has demonstrated its utility in various domains, including marketing, public opinion monitoring, and brand management. By extracting sentiments from tweets in real-time, businesses and organizations can make informed decisions, adapt strategies, and respond proactively to emerging trends.

11.1.3 Challenges Addressed

The development process involved overcoming challenges such as contextual understanding, handling sarcasm, and addressing biases in the sentiment analysis model. Techniques employed include leveraging pretrained models, implementing robust error handling, and refining the model based on ongoing feedback.

11.2 Future Scope

11.2.1 Integration of Advanced Models

As sentiment analysis technology evolves, future iterations of the Twitter Sentiment Analysis application could benefit from integrating more advanced models, such as transformer-based architectures like BERT or fine-tuned models specific to certain domains.

11.2.2 Multimodal Sentiment Analysis

Expanding the application to incorporate multimodal sentiment analysis, considering not only text but also images and potentially audio, could provide a more holistic understanding of sentiment expression on Twitter.

11.2.3 Real-Time Visualization and Reporting

Enhancements to the user interface, including real-time sentiment visualizations, sentiment trends over time, and sentiment breakdowns by location, could further elevate the application's usability for end-users.

11.2.4 Customization for Industry-Specific Insights

Tailoring the sentiment analysis model for specific industries or domains could amplify the application's impact. Customization may involve training the model on industry-specific datasets to improve accuracy and relevance.

11.2.5 Collaboration with Social Media Analytics Platforms

Exploring collaborations with social media analytics platforms or Twitter itself could open avenues for more extensive data access and integration, leading to a richer and more comprehensive sentiment analysis experience.

11.3 Conclusion and Acknowledgments

In conclusion, the Twitter Sentiment Analysis application presented in this report showcases the potential of sentiment analysis in extracting actionable insights from social media data. The journey from understanding methodologies to practical implementation has provided valuable insights into the complexities and opportunities within the field.

Acknowledgments are extended to the developers, researchers, and organizations contributing to the advancements in sentiment analysis technology. As we navigate the evolving landscape of natural language processing, the potential for impactful applications like Twitter Sentiment Analysis remains vast.

This report serves as a testament to the continuous evolution of sentiment analysis, and it is with anticipation and enthusiasm that we look forward to the future developments, innovations, and applications that will shape the field in the years to come.

References :

[1] https://www.lexalytics.com/news/lexalytics-unveils-sentiment-analysis-of-emoticons-acronyms/

[2] https://textblob.readthedocs.io/en/dev/classifiers.html

[3] https://stackoverflow.com/questions/48860422/text-blob-naive-bayes-classification

[4] https://chat.openai.com/

[5] https://www.kaggle.com/code/anoopjohny/threads-dataset-analysis/input

[6] Z. Yao, J. Yang, J. Liu, M. Keith, and C. Guan, "Comparing tweet sentiments in megacities using machine learning techniques: In the midst of COVID-19", Cities, Vol. 116, p. 103273, 2021.

[7] Z. Long, R. Alharthi, and A. E. Saddik, "Needfull–A tweet analysis platform to study human needs during the COVID-19 pandemic in New York state", IEEE

[8] O. Gencoglu and M. Gruber, "Causal Modeling of Twitter Activity During COVID-19", arXiv preprint arXiv:2005.07952, 2020

[9] H. Rosenberg, S. Syed, and S. Rezaie, "The Twitter pandemic: The critical role of Twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic", Canadian Journal of Emergency Medicine, Vol. 22, No. 4, pp. 418-21, 2020.

[10] Zhang, Y., Jin, R. and Zhou, Z. (2010). Understanding bag-of-words model: a statistical framework. International Journal of Machine Learning and Cybernetics, 1(1-4), pp.43-52.

[11] Park, C. and Seo, D. (2018). Sentiment analysis of Twitter corpus related to artificial intelligence assistants. 2018 5th International Conference on Industrial Engineering and Applications (ICIEA). El Rahman, S., AlOtaibi, F. and AlShehri, W. (2019). Sentiment Analysis of Twitter Data. 2019 International Conference on Computer and Information Sciences (ICCIS).

[12] Goyvaerts, J. (2006). Regular Expressions The Complete Tutorial.

[13] Kiruthika, Sanjana, & Giri, " Sentiment analysis of Twitter Data", International Journal of Innovation in Engineering and Technology, Vol No.6, April 2016

[14] Mitali Desai, Mayuri A. Mehta, "Techniques for Sentiment Analysis of Twitter Data: A Comprehensive Survey", International Conference on Computing, Communication, and Automation (ICCCA2016) [6] Sanjeev Kumar Sharma, "Sentiment Analysis: An analysis on its past, present and future scope", International Journal of advanced research in Science and Engineering, Vol.No.6, Issue No.07, July 2017.