TWITTER SENTIMENT ANALYSIS

A PROJECT REPORT

Submitted in partial fulfilment of the requirements for the

Award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

Submitted By

Priyanshu Saini (19EMCCS090)

Lokesh Kumar Gupta (19EMCCS066)

Manish Bhatnagar (20EMCCS203)

Nitin Jangid (20EMCCS204)

Yogesh Kumar (19EMCCS123)

Under the Supervision of

Mr. Arvind Sharma

(Assistant Professor)

BIKANER TECHNICAL UNIVERSITY, BIKANER

MODERN INSTITUTE OF TECHNOLOGY & RESEARCH CENTRE

ALWAR

April, 2023 (2022-23)

BIKANER TECHNICAL UNIVERSITY, BIKANER

CERTIFICATE

Certified that this project report **"TWITER SENTIMENT ANALYSIS USING NLP"** is the original work of "**PRIYANSHU SAINI, LOKESH KUMAR GUPTA, MANISH BHATNAGAR, NITIN JANGID, YOGESH KUMAR"** students of B. Tech. Final Year VIII Semester *(Computer Science and Engineering)* who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. J.R.Arun Kumar	Mr.Arvind Sharma
HEAD OF THE DEPARTMENT	**SUPERVISOR Assistant Professor**
Computer Science Department	Computer Science Department
MITRC College, Alwar	MITRC College, Alwar

ACKNOWLEDGEMENT

Firstly, we would like to express our gratitude to our advisor for the beneficial comments and remarks.

We express our sincere thanks to Prof. **S. K. Sharma (Director)** of the Modern Institute of Technology & Research Centre, Alwar.

We pay our deep sense of gratitude to **Dr. J.R Arun Kumar**, Head of the Computer Science & Engineering Department of Modern Institute of Technology & Research Centre, Alwar for encouraging us to the highest peak and providing us the opportunity to present the Project.

We acknowledge a deep sense of gratitude to **Mr. Arvind Sharma** (ProjectSupervisor), Department of Computer Science & Engineering. Modern Institute of Technology & Research Centre, Alwar (Rajasthan) for her constant support and guidance during this work. His honesty, thoroughness, and perseverance have been a constant source of inspiration for us. We would like to thank our institute and faculty members without them this project would

have been a distant reality, we also extend our heartfelt thanks to our families and well-wishers.

Finally, but not least, our parents are also an important inspiration for us. With due respect, we express our gratitude to them.

Priyanshu Saini (19EMCCS090)

Lokesh Kumar Gupta(19EMCCS066)

Manish Bhatnagar (20EMCCS203)

Nitin Jangid (20EMCCS204)

Yogesh Kumar(19EMCCS123)

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGENO
	CERTIFICATE	i
	ACKNOWLEDGEMENT	ii
	TABLE OF CONTENT	iii-v
	LIST OF FIGURE	vi
	ABSTRACT	vii
1.	**INTRODUCTION**	1-5
	1.1 Problem Description	2
	1.2 Motivation	3-4
	1.3 Objective	5
2.	**LITERATURE REVIEW**	6-8
3.	**SENTIMENT ANALYSIS**	9-11
	3.1 Definition	9
	3.2 Level	9-10
	3.2.1 Document	9
	3.2.2 Sentence	10
	3.2.3 Entity or Aspect	10
	3.3 Approaches	10-11
	3.3.1 Keyword Spotting	10
	3.3.2 Lexical affinity	10
	3.3.3 Statistical method	11
	3.4 Features	11
	3.4.1 Terms presence and frequency	11
	3.4.2 Parts of speech	11
	3.4.3 Opinion words and phrases	11

4.	**SOLUTION APPROACH**	12-18
5.	**METHODOLOGY**	19-28
	5.1 Data collection and preprocessing	19-22
	5.1.1 Description of the data source & collection process	19-20
	5.1.2 Data cleaning and normalization techniques	20-21
	5.1.3 Feature extraction and selection	21-22
	5.2 Sentiment classification model	22-25
	5.2.1 Description of the model architecture	22-23
	5.2.2 Algorithm selection and justification	23-24
	5.2.3 Training and evalution process	24-25
	5.3 Visualization and interpretation of results	25-28
	5.3.1 Distribution of sentiment across the dataset	25-26
	5.3.2 Correlation analysis with external factors	26-27
	5.3.3 Qualitative analysis of misclassified tweets	27-28
	5.4 Technology Stack	28
	5.4.1 Hardware Requirements	28
	5.4.2 Software Requirements	28
6.	**IMPLEMENTATION**	29-34
	6.1 Tools and Technologies	29-32
	6.1.1 Python	29-30
	6.1.2 Tweepy	30
	6.1.3 NLTK	31-32
	6.2 How to do Twitter Sentiment Analysis	32-34

7.	**RESULT ANALYSIS**	35-38
	7.1 User Registration	35-36
	7.2 Analysis Page	37-38
8.	**CONCLUSION AND FUTURE SCOPE**	39-40
	REFERENCES	41

LIST OF FIGURES

FIGURE NO.	NAME	PAGE NO.
4	Flow chart of twitter sentiment analysis	13
5.1	Data Pre-processing	22
5.2	Model of proposed system	28
6.1	NLTK Text Pre-processing	32
6.2	Insight Visualization	34
7.1	User Registration	35
7.1.1	Login Page	36
7.1.2	Home Page	36
7.2	Analysis Page	37
7.2.1	Result	38
7.2.2	Result Pie Chart	39

ABSTRACT

In this challenging world, social media plays a vital role as it is at the pinnacle of data sharing. The advancement in technology has made a huge amount of information available for data analysis and it is on the hotlist nowadays. Opinions of the people are expressed and shared across various social media platforms like Twitter, Facebook, and Instagram.

Twitter is a prodigious platform containing an ample amount of data and analyzing the data is of topmost priority. One of the most widely utilized approaches for classifying an individual's emotions displayed in subjective data is sentiment analysis.

Sentiment analysis is done using various algorithms of machine learning like Support Vector Machine, Naive Bayes, Long Short-Term Memory, Decision Tree Classifier, and many more, but this paper aims at the generalized way of performing Twitter sentiment analysis using flask environment.

Flask environment provides various inbuilt functionalities to analyze the sentiments of text into three different categories: positive, negative, and neutral. Also, it makes API calls to the Twitter Developer account to fetch the Twitter data. After fetching and analyzing the data, the results get displayed on a webpage containing the percentage of positive, negative, and neutral tweets for a phrase in a pie chart. It displays the language analysis for the same phrase.

Furthermore, the webpage calls attention to the tweets done on that phrase and reveals the details of the tweets. Considering the major industry runners of three different sectors namely Enterprises, Sports Apparel Industry, and Multimedia Industry, we have analyzed and compared sentiments of two different Multinational companies from each sector.

Chapter - 1

INTRODUCTION

Twitter sentiment analysis is a technique used to extract insights from the opinions and emotions expressed in tweets. It involves analyzing the text of tweets to determine whether they express positive, negative, or neutral sentiment towards a particular topic, brand, product, or event. Twitter has become a valuable source of data for sentiment analysis due to its popularity, real-time nature, and the availability of public APIs.

Sentiment analysis has various applications, such as brand monitoring, customer service, political analysis, and market research. For example, a company might use sentiment analysis to monitor the public perception of their brand on Twitter, and adjust their marketing strategy accordingly. A political campaign might use sentiment analysis to understand how the public feels about certain issues, and tailor their messaging accordingly. A researcher might use sentiment analysis to study the emotional impact of a natural disaster on Twitter users.

There are several methods for performing sentiment analysis on Twitter data, such as rule-based methods, machine learning algorithms, and deep learning models. Rule-based methods involve creating a set of rules to identify sentiment based on the presence of certain words or phrases. Machine learning algorithms involve training a model on a labelled dataset to classify the sentiment of the tweet. Deep learning models use neural networks to perform sentiment analysis.

One of the challenges of performing sentiment analysis on Twitter data is the noisy nature of the data. Twitter data often contains slang, abbreviations, misspellings, and other types of noise that can make it difficult to accurately classify sentiment. Another challenge is the need to account for context and sarcasm, which can change the meaning of a tweet.

In contrast, consumers have all the power when it comes to what consumers want to see and how consumers respond. With this, the company's success & failure is publicly shared and end up with word of mouth. However, the social network can change the behavior and decision making of consumers, for example, mentions that 87% of internet users are influenced in their purchase and decision by customer's review. So that, if organization can catch up faster on what their customer's think, it would be more beneficial to organize to react on time and come up with a good strategy to compete their competitors.

In conclusion, Twitter sentiment analysis is a technique used to analyze the sentiment expressed in tweets towards a particular topic, brand, product, or event. It has various applications in areas such as brand monitoring, customer service, political analysis, and market research. The choice of approach depends on the specific problem we are trying to solve and the resources available to us.

Sentiments of tweets can be categorized into many categories like positive, negative, neutral, extremely positive, extremely negative, and so on. The two types of sentiments considered in this classification experiment are positive and negative sentiments. The data, being labelled by humans, has a lot of noise, and its hard to achieve good accuracy. Currently, the best results are obtained by Support Vector Machine(SVM) for a feature set containing stemming, and Bigram that gives an accuracy of 82.55. The main algorithms used in this project are SVM and Naive bayes.

Sentiment analysis of public is highly critical in macro-scale socioeconomic phenomena like predicting the stock market rate of a particular firm. This could be done by analyzing overall public sentiment towards that firm with respect to time and using economics tools for finding the correlation between public sentiment and the firm's stock market value. Firms can also estimate how well their product is responding in the market, which areas of the market is it having a favorable response and in which a negative response (since twitter allows us to download stream of geo-tagged tweets for particular locations.

These are the steps used in Machine Learning for analyzing the sentiment.

- Stemming
- Tokenization
- Part of speech tagging
- Parsing
- Lexicon analysis (depending on the relevant context)

1.1 Problem description

The problem of Twitter sentiment analysis involves analyzing the sentiment expressed in tweets towards a particular topic, brand, product, or event. The main challenge in this problem is the noisy nature of the data, which includes slang, abbreviations, misspellings, and other types of noise that can make it difficult to accurately classify sentiment. Additionally, tweets often contain sarcasm, irony, and other forms of figurative language that require context to understand the true sentiment being expressed.

The problem of Twitter sentiment analysis has various applications, such as brand monitoring, customer service, political analysis, and market research. For example, a company might use sentiment analysis to monitor the public perception of their brand on Twitter, and adjust their marketing strategy accordingly. A political campaign might use sentiment analysis to understand how the public feels about certain issues, and tailor their messaging accordingly. A researcher might use sentiment analysis to study the emotional impact of a natural disaster on Twitter users.

The problem of Twitter sentiment analysis can be addressed using various methods, such as rule-based methods, machine learning algorithms, and deep learning models. Rule-based methods involve creating a set of rules to identify sentiment based on the presence of certain words or phrases. Machine learning algorithms involve training a model on a labelled dataset to classify the sentiment of the tweet. Deep learning models use neural networks to perform sentiment analysis.

In summary, the problem of Twitter sentiment analysis involves analyzing the sentiment expressed in tweets towards a particular topic, brand, product, or event. This problem has various applications in areas such as brand monitoring, customer service, political analysis, and market research. The noisy and context-dependent nature of the data makes this problem challenging, but there are various methods that can be used to address it.

1.2 Motivation

The motivation behind Twitter sentiment analysis is to gain insights into the opinions and emotions expressed by users on Twitter towards a particular topic, brand, product, or event. Twitter has become a valuable source of data for sentiment analysis due to its popularity, real-time nature, and the availability of public APIs.

By performing sentiment analysis on Twitter data, we can gain a better understanding of how people feel about a particular topic, brand, product, or event, and use this information to make informed decisions. For example, a company might use sentiment analysis to monitor the public perception of their brand on Twitter, and adjust their marketing strategy accordingly. A political campaign might use sentiment analysis to understand how the public feels about certain issues, and tailor their messaging accordingly. A researcher might use sentiment analysis to study the emotional impact of a natural disaster on Twitter users.

Twitter sentiment analysis has various applications, such as brand monitoring, customer service, political analysis, and market research. It can help companies and organizations understand their customers better, identify potential issues, and improve their products and services. It can also help researchers gain insights into the emotional impact of events and phenomena.

The motivation behind Twitter sentiment analysis is to gain insights into the opinions and emotions expressed by users on Twitter towards a particular topic, brand, product, or event, and use this information to make informed decisions. Twitter sentiment analysis has various applications in areas such as brand monitoring, customer service, political analysis, and market research.

There are some data that might help illustrate the motivation behind Twitter sentiment analysis.

According to a report by Hootsuite, there are over 330 million monthly active users on Twitter, and over 500 million tweets are sent out every day. This makes Twitter a rich source of data for sentiment analysis.

In addition, a study by Twitter and Market Probe International found that 54% of users on Twitter have taken action after seeing a brand mentioned in a tweet. This highlights the importance of monitoring and understanding the public perception of a brand on Twitter.

Another study by Twitter and Research Now found that 60% of users on Twitter have used the platform to learn about news and events, and 40% of users have shared their opinions about news and events on Twitter. This suggests that Twitter sentiment analysis can be useful for understanding public opinion on a wide range of topics.

Overall, these statistics suggest that Twitter sentiment analysis can provide valuable insights into public opinion and help companies and organizations make informed decisions.

1.3 Objective

The objectives for Twitter sentiment analysis are to:

1. Monitor brand reputation: Companies and organizations can use sentiment analysis to monitor public opinion about their brand on Twitter, and identify potential issues that could damage their reputation.

2. Understand customer feedback: Sentiment analysis can help companies understand how customers feel about their products or services, and identify areas for improvement.

3. Analyze market trends: Sentiment analysis can be used to track market trends and identify emerging topics of interest to customers.

4. Gauge public opinion: Twitter sentiment analysis can provide insights into public opinion on a wide range of topics, such as politics, social issues, and current events.

5. Identify influencers: Sentiment analysis can help companies identify influential Twitter users who could help promote their brand or product.

6. Improve customer service: By monitoring Twitter sentiment, companies can identify customer complaints or issues in real-time, and respond quickly to resolve them.

7. Predict outcomes: Sentiment analysis can be used to predict future outcomes based on public opinion, such as the success of a product launch or the outcome of an election.

Overall, the objectives of Twitter sentiment analysis are to gain insights into public opinion, identify potential issues, and use this information to make informed decisions.

Chapter – 2

LITERATURE REVIEW

It is of no surprise that Twitter can also be responsible for commendation or defamation of a brand or company since it is very convenient for users to post their personal liking and preference in the form of online reviews.

1. "A review of sentiment analysis research in Twitter" by Muhammad M. Hossain and Muhammad A. Hussain., (2019): This review provides a comprehensive overview of sentiment analysis techniques used in Twitter. The authors cover various approaches, including lexicon-based methods, machine learning methods, and hybrid methods. They also discuss the challenges and limitations of these techniques, such as handling sarcasm, irony, and context. The authors also identify research gaps and future directions for Twitter sentiment analysis, such as addressing the issue of bias in sentiment analysis and developing better models for handling linguistic nuances.

2. "A comprehensive review on sentiment analysis and opinion mining techniques for social media analysis" by Mohammad Salim Beg and Tarek R. Besold., (2020): This review provides an overview of sentiment analysis and opinion mining techniques used for social media analysis, including Twitter. The authors cover various approaches, such as lexicon-based methods, machine learning methods, and deep learning methods. They also discuss the challenges and limitations of these techniques, such as data sparsity and noisy data. The authors identify potential research directions, such as developing more effective methods for handling social media data and integrating sentiment analysis with other NLP tasks.

3. "Twitter sentiment analysis: A survey of the state-of-the-art" by Alok Ranjan Pal and Asif Ekbal., (2020): This review provides a comprehensive survey of the state-of-the-art in Twitter sentiment analysis. The authors cover various approaches, such as lexicon-based methods, machine learning methods, and deep learning methods. They also discuss the challenges and future directions for Twitter sentiment analysis, such as handling linguistic nuances and developing more effective methods for handling noisy data. The authors identify potential research directions, such as developing more effective models for multi-lingual sentiment analysis and exploring the use of sentiment analysis in areas such as health and finance.

4. "A review of deep learning methods for sentiment analysis in social media" by Xiangyu Wang and Enhong Chen., (2019): This review provides a comprehensive overview of deep learning methods used for sentiment analysis in social media, including Twitter. The authors cover various approaches, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and attention mechanisms. They also discuss the strengths and limitations of these methods, such as the need for large amounts of labelled data. The authors identify potential research directions, such as exploring the use of multi-task learning and transfer learning for sentiment analysis in social media.

5. "A systematic review of Twitter sentiment analysis: An insight into the world of emotions" by Shweta Gupta and Khushboo Yadav., (2019): This review provides a systematic analysis of Twitter sentiment analysis research. The authors cover various aspects of Twitter sentiment analysis, such as data collection, pre-processing, and classification. They also discuss the limitations and challenges of sentiment analysis on Twitter, such as handling sarcasm and irony, and the need for domain-specific sentiment lexicons. The authors identify potential research directions, such as developing better models for detecting sentiment in multimodal content and exploring the use of sentiment analysis for political and social applications.

6. "A review of sentiment analysis methods and applications in Twitter" by Narendra Choudhary, Ahmed Raza, and Yousuf Al-Habsi., (2021): This review provides an overview of sentiment analysis methods and applications in Twitter. The authors cover various approaches, such as lexicon-based methods, machine learning methods, and deep learning methods. They also discuss the challenges and limitations of these techniques, such as handling noisy data and developing models for handling linguistic nuances. The authors identify potential research directions, such as developing more effective methods for analyzing sentiment in multilingual tweets and exploring the use of sentiment analysis for applications such as customer feedback analysis.

7. "Twitter sentiment analysis: A review of methods and applications" by Aya Khalaf and John Kelleher., (2021): This review provides a comprehensive overview of Twitter sentiment analysis methods and applications. The authors cover various approaches, such as lexicon-based methods, machine learning methods, and deep learning methods. They also discuss the challenges and limitations of these techniques, such as handling sarcasm

and irony, and the need for domain-specific sentiment lexicons. The authors identify potential research directions, such as exploring the use of ensemble methods for sentiment analysis and developing better models for handling multilingual tweets.

Overall, these literature reviews provide valuable insights into the current state-of-the-art in Twitter sentiment analysis and identify potential research directions for future work.

8. "A review of sentiment analysis methods and applications in Twitter" compared the accuracies of the following models by Bhumika Gupta, Monika Negi (2017): DAN2, SVM, Bayesian Logistic Regression, Naïve Bayes, Random Forest Classifier, Neural Network, Maximum Entropy and Ensemble classifier. The last two classifiers gave the highest performance rate. They also concluded that the efficiency of classifier is inversely proportional to the number of classes made.

9. "**A review of sentiment analysis methods and applications in Twitter" by Ankit Pradeep Patel, Ankit Vithalbhai Patel, Sanjaykumar Ghanshyambhai Butani, Prashant B. Sawant (2017):** This review provided SVM outperforms ANN in text categorization. They have expressed the need to use supervised algorithm to analyse the sentiments instead of the Vector Quantization which comes under the category of unsupervised algorithm.

10. "**A review of sentiment analysis methods and applications in Twitter" by Akshi Kumar, Teeja Mary Sebastian (2012) :** The aggregation of the positive/negative values of the sentences by combining two methods, namely corpus based (for adjectives) and dictionary based (for verbs and adverbs).

Chapter - 3

SENTIMENT ANALYSIS

3.1 Definition

Sentiment and Subjectivity are mainly context and domain dependent. Not only the changes in vocabulary are the reason behind that but one more reason is the dual meaning or sentiments of same expression in different domains. Consider the example of expression 'go and read the book'. In case of book reviews this expression gives the positive polarity about the product but in case of movie review the same expression gives negative polarity about the product. Sentiment Analysis is more focused on extraction of polarity about a particular topic rather than assigning a particular emotion to the text. Opinion Mining and Sentiment Analysis are the branches of Text Mining which refer to the process of extracting nontrivial patterns and interesting information from unstructured script documents. We can say that they are the addition to data mining and knowledge discovery. Opinion Mining and Sentiment Analysis focus on polarity detection and emotion recognition correspondingly. Opinion Mining has more marketable potential higher than data mining as it the most natural form of storing the information in text format. It is much complex task than data mining because it has to deal with unstructured and fuzzy data. It is a multi-disciplinary area of research because it constitutes adoption of techniques in information retrieval, text analysis and extraction, auto-categorization, machine learning, clustering, and visualization. Though Sentiment Analysis and Opinion Mining might look the same as the fields like traditional text mining or fact-based analysis, it varies because of following facts. Sentiment Classification is the binary polarity classification which deals with a relatively small number of classes. Sentiment classification is easy task compared to text auto-categorization.

3.2 Levels

We can divide sentiment analysis in following levels.

3.2.1 Document

The task at this level is classifying the sentiment for document. The document is on single topic is considered. Thus, texts which comprise comparative learning cannot be considered under this level.

3.2.2 Sentence

The task at this level goes to the sentences; it determines whether each sentence expresses a positive opinion, negative opinion, or neutral view. If a sentence states no opinion means it is a neutral. This level of analysis is closely related to subjectivity classification. The subjective statement displays the polarity of an entity in affirmative-negative terms i.e., good-bad terms. Hence it is easy to obtain sentiment from it. But Objective statement does not give separation directly by affirmative negative terms. These are abstract sentences which are fact based.

3.2.3 Entity or Aspect

Aspect level gives detailed analysis. The core task of entity level is to identification of aspect of the text.

For example, in a review of mobile if a customer says," Sound is good but the handset is not handy." In this the aspect is sound and handiness. Here sentiment analysis becomes two level tasks i.e., finding the aspects in the text and then classifying in respective aspect. Aspect level sentiment analysis is superior to Document and Sentence level sentiment analysis. Sentiment analysis of topic or body which may or may not be hidden in the document is done. Thus, comparative statements are also part of entity level sentiment analysis.

3.3 Approaches

We can do opinion mining and sentiment analysis in following ways: keyword spotting, lexical affinity, statistical methods.

3.3.1 Keyword Spotting

In this technique the text is categorized based on the presence of fairly unambiguous words present in it. Thus, the words or keywords present in the text have the importance with respect to sentiment analysis.

3.3.2 Lexical affinity

For a particular emotion, Lexical affinity assigns arbitrary words a probabilistic similarity.

3.3.3 Statistical methods

It calculates the valence or target of affective keywords and word co-occurrence frequencies on the base of a large training corpus. In early work it was aim to classify entire document into overall affirmative or negative. These systems mainly depend on supervised learning approaches which depend on manually labelled data. The examples of such systems are movie or product review databases. Many times, sentiments are not restricted to document level texts. It can be extracted from sentence level text. In such cases sentiment analysis can be done using detected opinion-bearing lexicon items. Or sentiments are not limited to particular target, they can contrary towards same topic or multiple topics can be present in the same document.

3.4 Features

Sentiment features are as follows:

3.4.1 Terms presence and frequency

These features are nothing but individual words or word n-grams and their frequency counts. It either uses the term frequency weights or gives binary weighting to the words.

3.4.2 Parts of speech (POS)

It set up finding adjectives from the text, as they are important indicators of opinions.

3.4.3 Opinion words and phrases

These words themselves express opinion about the product or service in the text. For e.g., good or bad, like or hate. Some phrases also express opinions without using opinion words.

Negations: the presence of negative words may change the opinion orientation like not good is equivalent to bad.

Chapter - 4

SOLUTION APPROACH

Twitter sentiment analysis involves the use of computational techniques to identify and extract subjective information from text data, such as tweets. Sentiment analysis can help businesses and organizations to better understand their customers' needs, preferences, and sentiments towards their products or services. It can also be used for political analysis, where it can help to understand public opinion and sentiment towards political candidates or issues.

Sentiment analysis, also known as opinion mining, is a computational technique used to identify and extract subjective information from text data. In the context of Twitter, sentiment analysis involves extracting and analyzing tweets to determine the overall sentiment or attitude of the text, whether it is positive, negative, or neutral.

There are several approaches to sentiment analysis, including rule-based, lexicon-based, and machine learning-based methods.

Rule-based methods involve the use of handcrafted rules and heuristics to identify sentiment in text. These methods are often based on linguistic patterns and rules, such as the presence of certain words or phrases. While rule-based methods can be effective for certain tasks, they can be limited in their ability to generalize to new data and may not capture more subtle nuances of sentiment.

Lexicon-based methods, on the other hand, rely on pre-existing dictionaries of words and phrases that have been labeled with their associated sentiment. These methods use the presence and frequency of positive and negative words in a text to determine the overall sentiment. While lexicon-based methods can be effective, they may not be able to capture the complexity and nuance of sentiment in text.

Machine learning-based methods involve training models on labeled datasets to predict the sentiment of new text. These methods can be highly effective and can capture the subtleties of sentiment in text. Common machine learning algorithms used for sentiment analysis include Naive Bayes, Support Vector Machines (SVMs), and Recurrent Neural Networks (RNNs).

Once sentiment has been identified in Twitter data, it can be used for a variety of applications, including brand monitoring, market research, and political analysis. However, it is important to note that sentiment analysis is not a perfect tool and can be influenced by factors such as sarcasm, irony, and cultural context.

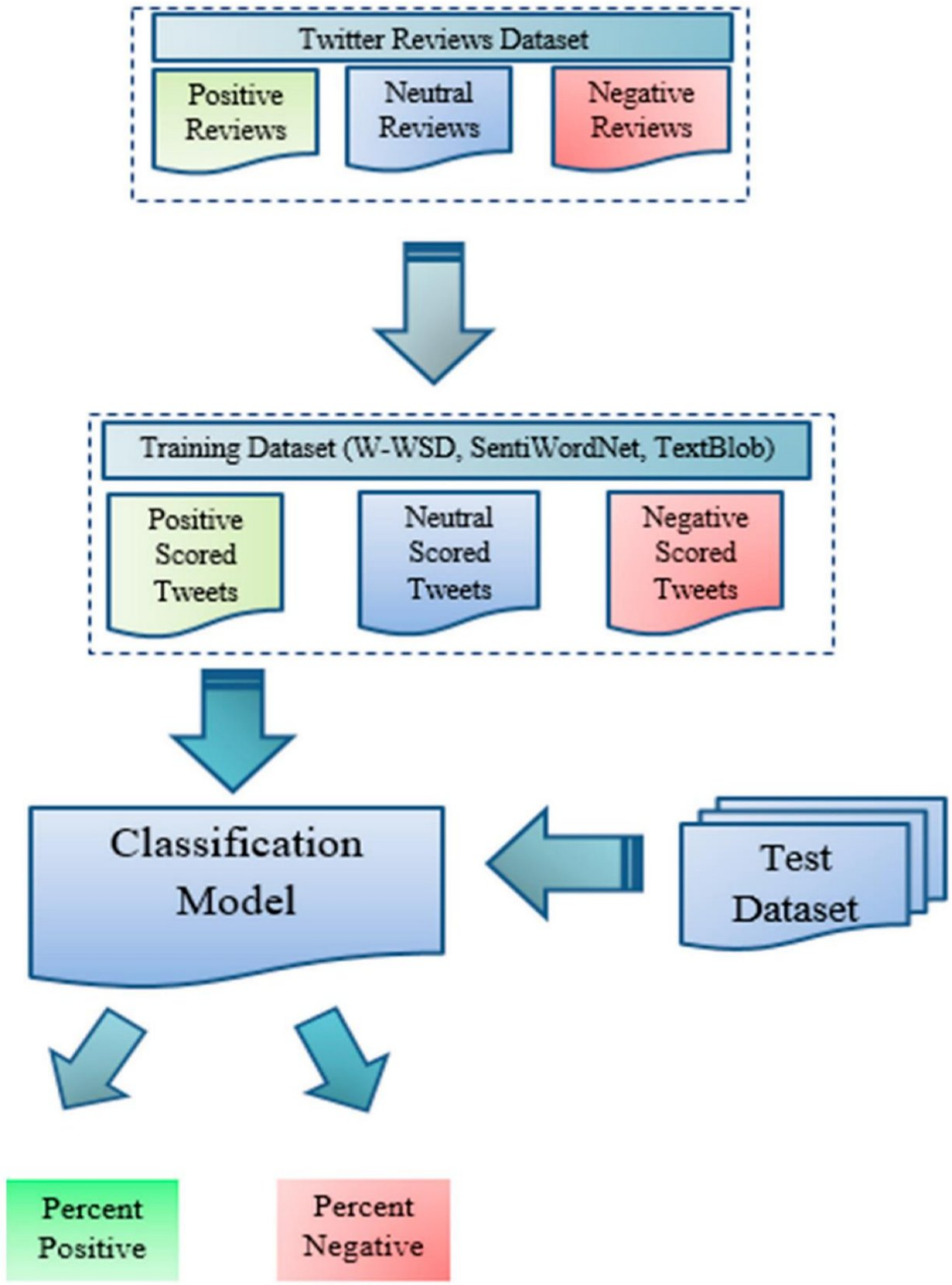

Fig 4: Flow chart of twitter sentiment analysis

.

The solution approach for Twitter sentiment analysis using natural language processing.

1. Data Collection and Preprocessing

Data collection for Twitter sentiment analysis can be performed using Twitter's API or third-party tools such as web scrapers or data providers. Once the data has been collected, it must be preprocessed to prepare it for analysis. This involves removing irrelevant information such as URLs and usernames, as well as removing stop words (common words such as "the" or "and") and performing tokenization, stemming, and lemmatization. Tokenization involves breaking the text into individual words or tokens, while stemming and lemmatization involve reducing words to their base form to ensure consistency in the dataset. Data cleaning and normalization techniques such as these are important for improving the quality of the data and ensuring that the analysis is accurate.

2. Feature Extraction and Selection

Feature extraction involves converting the text data into a numerical format that can be used by machine learning algorithms. The bag-of-words model is a common approach, which involves representing each document as a vector of word frequencies. Another approach is TF-IDF, which takes into account the frequency of each word in the document and across the entire corpus to assign a weight to each word. Word embeddings are another popular technique, which represent each word as a dense vector in a high-dimensional space. Feature selection involves identifying the most relevant features for classification, which can improve the efficiency and accuracy of the algorithm.

3. Algorithm Selection and Training

There are several machine learning algorithms that can be used for sentiment analysis, including Naive Bayes, Support Vector Machines, and Neural Networks. The choice of algorithm depends on the specific requirements of the project and the characteristics of the dataset. Once the algorithm has been selected, the model must be trained on a labeled dataset. The labeled dataset should be representative of the data that the model will be applied to and should contain a sufficient number of examples for the algorithm to learn from. Training involves adjusting the model's parameters to minimize the error on the training data.

4. Evaluation

Once the model has been trained, it must be evaluated on a held-out dataset to assess its performance. Evaluation metrics such as accuracy, precision, recall, and F1 score are commonly used to measure the performance of the model. ROC curves and confusion matrices can also be used to visualize the performance of the model. The evaluation results can be used to identify areas where the model is performing poorly and to fine-tune the model for better performance.

Overall, the solution approach for Twitter sentiment analysis using natural language processing involves several steps, each of which plays an important role in developing an accurate and reliable sentiment analysis model. By following this approach and applying the appropriate techniques and algorithms, we can extract insights from social media data and make informed decisions based on the sentiment of the public.

Deep Learning is computationally expensive and do not do well (poor performance and slow convergence) with high-dimensional, sparse vectors. When we extract features from the original text for model training, we need to represent them as vectors that are dense. One such technique is to convert each text into a sequence of numbers, where each number is mapped to a word in the vocabulary. Taking it a step further, we need to map words that have similar usage/meaning to similar real number vectors (rather than an index) using Word Embeddings. Without Word Embeddings, the model will misinterpret the index number of the words as having significance. Word Embeddings put all words into a multi-dimensional vector space so their similarities can be measured by distance. Generating Word Embeddings is done using open source pre-trained models (such as Word2vec, GloVe, or fastText) or custom neural network (unsupervised learning) models. When training custom Word Embeddings, it can be done separately or jointly with the neural network model of the task at hand (as an addition layer). This is the approach we take as it tends to lead to embeddings that are targeted to the context of data as well as the objective. In contrast to a document-term matrix that is sparse (hundreds of thousands of dimension), using Word Embeddings produces vectors that have fewer (hundreds) dimensions while capturing semantic similarities. Word Embeddings is considered one of the key breakthroughs of Deep Learning for challenging NLP problems.

While Rule-Based solutions have proved to be performant and reliable, it is rigid and limited by a lexicon that could expire. Manually creating and validating a comprehensive sentiment lexicon is difficult. Instead, we want to move towards a Machine Learning solution that associates input texts to corresponding tags algorithmically. During the training process, pairs of texts and tags (sentiments) are fed into the Machine Learning algorithm to create a model capable of making predictions on new texts. An important caution worth mentioning is that supervised Machine Learning requires data to be labelled, which can mean that any subjectivity and bias in the data would be reflected in the model. Compared to pre-trained models, custom models give more control over the output and are suitable for applications that are specific.

The Naive Bayes classifier uses probabilities to make predictions based on prior knowledge of conditions that might be related. In other words, it uses conditional probabilities of each lexical feature occurring in either positive or negative text in the training data to arrive at the outcome.

Machine Learning for NLP requires preprocessing to extract features from the original text(s). Specifically, it requires transformation of text data to numerical representations before the algorithms can process them, i.e. vectorization. For Naive Bayes, we will construct a simple DTM (document-term matrix) for model consumption although additional features can be included such as length of text, time/location of publication, named entities, etc. DTM tends to result in a wide feature space as each unique word or phrase in the vocabulary of the entire corpus becomes a feature. We include some data cleaning steps to help with dimensionality reduction and model (classifiers) performance. More about data preprocessing for NLP can be found here. The vector representation is constructed by counting the TF (Term Frequency) and weighting them with IDF (Inverse Document Frequency). We can use N-grams (consecutive terms) to capture some context in the text such as negations. N-gram contexts are a blunt instrument and can't always capture expressions correctly. They can also negatively impact the model by burdening it with too many features.

Lexicon-based Approach Lexicon-based methods make use of predefined list of words where each word is associated with a specific sentiment. The lexicon methods vary according to the context in which they were created and involve calculating orientation for

a document from the semantic orientation of texts or phrases in the documents. Besides, also states that a lexicon sentiment is to detect word-carrying opinion in the corpus and then to predict opinion expressed in the text has shown the lexicon methods which have a basic paradigm which are: i. Preprocess each tweet, post by remove punctuation ii. Initialize a total polarity score (s) equal 0 -> s=0 iii. Check if token is present in a dictionary, then If token is positive, s will be positive (+) If token is negative, s will be negative (-) iv. Look at the total polarity score of tweet post If s > threshold, tweet post as positive If s < threshold, tweet post as negative However, highlighted one advantage of leaning-based method, is that it has the ability to adapt and create trained models for specific purposes and contexts. In contrast, an availability of labeled data and hence the low applicability of the method of new data which is cause labeling data might be costly or even prohibitive for some tasks

Machine learning methods often rely on supervised classification approaches where sentiment detection is framed as a binary which are positive and negative. This approach requires labeled data to train classifiers . This approach, it becomes apparent that aspects of the local context of a word need to be taken into account such as negative (e.g. Not beautiful) and intensification (e.g. Very beautiful). However, showed a basic paradigm for create a feature vector is:

i. Apply a part of speech tagger to each tweet post
ii. Collect all the adjective for entire tweet posts
iii. Make a popular word set composed of the top N adjectives
iv Navigate all of the tweets in the experimental set to create the following:
    - Number of positive words
    - Number of negative words
    - Presence, absence or frequency of each word.

showed some example of switch negation, negation simply to reverse the polarity of the lexicon: changing beautiful (+3) into not beautiful (-3). More examples: She is not terrific (6-5=1) but not terrible (-6+5=-1) either. In this case, the negation of a strongly negative or positive value reflects a mixed perspective which is correctly captured in the shifted value. However, has mentioned the limitation of machine-learning-based approach to be more suitable for Twitter than the lexical based method. Furthermore, stated that machine learning methods can generate a fixed number of the most regularly happening popular words which assigned an integer value on behalf of the frequency of the word in the Twitter.

Case-Based Reasoning (CBR) is one of the techniques available to implement sentiment analysis. CBR is known by recalling the past successfully solved problems and use the same solutions to solve the current closely related problems identified some of the advantages of using CBR that CBR does not require an explicit domain model and so elicitation becomes a task of gathering care histories and CBR system can learn by acquiring new knowledge as cases. This and the application of database techniques make the maintenance of large columns of information easier.

Artificial Neural Network (ANN) or known as neural network is a mathematical technique that interconnects group of artificial neurons. It will process information using the connections approach to computation. ANN is used in finding the relationship between input and output or to find patterns in data.

Alchemy API performs better than the others in terms of the quality and the quantity of the extracted entities . As time passed the PythonTwitter Application Programming Interface (API) is created by collected tweets. Python can automatically calculated frequency of messages being retweeted every 100 seconds, sorted the top 200 messages based on there-tweeting frequency, and stored them in the designated database. As the Python Twitter API only included Twitter messages for the most recent six days, collected the data needed to be stored in a different database.

Chapter - 5

METHODOLOGY

Certainly, here's a more detailed explanation of the methodology for conducting Twitter sentiment analysis using natural language processing:

5.1 Data collection and preprocessing

5.1.1 Description of the data source and collection process

In Twitter sentiment analysis, the data source is the tweets that are posted on the Twitter platform. Twitter provides an API that allows developers to extract tweets based on certain keywords, hashtags, or user handles. These APIs can be used to collect a large volume of tweets that are relevant to the topic of interest.

The collection process typically involves the following steps:

1. Identifying the topic of interest: The first step in data collection is to identify the topic of interest. This could be a specific event, product, or service.

2. Defining the search query: Once the topic of interest has been identified, we need to define the search query that will be used to extract the relevant tweets. The search query typically includes keywords, hashtags, or user handles that are related to the topic of interest.

3. Using the Twitter API: Twitter provides APIs that allow developers to extract tweets based on certain search parameters. The API provides access to both real-time and historical tweets.

4. Data sampling: In some cases, we may want to sample the tweets to ensure that we have a representative sample of the population. For example, we may want to sample tweets from a specific geographic region or time period.

5. Data filtering: Once the tweets have been collected, we may want to filter the data based on certain criteria such as language, user location, or user demographics.

6. Data preprocessing: After the data has been collected and filtered, we need to preprocess the data to remove any noise and irrelevant information. This involves several steps such

as tokenization, stop word removal, stemming/lemmatization, and handling negations, emojis, and emoticons.

Overall, the data collection process for Twitter sentiment analysis involves a combination of manual and automated techniques to extract, filter, and preprocess the tweets. The quality of the data collected is crucial for the accuracy of the sentiment analysis model.

5.1.2 Data cleaning and normalization techniques

Data cleaning and normalization are important steps in preparing the data for sentiment analysis. These steps help to remove noise and irrelevant information from the data and ensure that the data is in a consistent format. Here are some common data cleaning and normalization techniques used in Twitter sentiment analysis:

1. Removing special characters and punctuation: Twitter data often contains special characters and punctuation marks that are not relevant to sentiment analysis. These characters can be removed using regular expressions or string manipulation techniques.

2. Removing URLs and mentions: Tweets often contain URLs and user mentions that are not relevant to sentiment analysis. These can be removed using regular expressions or string manipulation techniques.

3. Lowercasing: Converting all text to lowercase can help to ensure consistency and reduce the number of unique words in the data.

4. Removing stop words: Stop words are commonly used words in a language that do not carry much meaning, such as "the" and "and". Removing these words can help to reduce noise in the data.

5. Stemming or lemmatization: Stemming and lemmatization are techniques used to reduce words to their base form. This helps to reduce the number of unique words in the data and improve the accuracy of the sentiment analysis.

6. Handling negations: Negations such as "not" or "no" can change the meaning of a sentence. These should be handled carefully to ensure that the sentiment analysis accurately reflects the intended meaning.

7. Handling emojis and emoticons: Emojis and emoticons are commonly used in Twitter data to convey emotions. These can be converted to text or removed from the data, depending on the needs of the analysis.

Overall, data cleaning and normalization are crucial steps in preparing the data for sentiment analysis. By removing noise and irrelevant information and ensuring consistency in the data, we can improve the accuracy of the sentiment analysis model.

5.1.3 Feature extraction and selection

Feature extraction and selection are important steps in preparing the data for sentiment analysis. These steps help to reduce the dimensionality of the data and extract the most relevant information for the analysis. Here are some common feature extraction and selection techniques used in Twitter sentiment analysis:

1. Bag-of-words: Bag-of-words is a commonly used technique for feature extraction in natural language processing. It involves creating a vocabulary of all unique words in the data and representing each document as a vector of word counts.

2. TF-IDF: Term frequency-inverse document frequency (TF-IDF) is a weighting scheme commonly used in information retrieval and text mining. It assigns weights to each term in a document based on how frequently it appears in the document and how rare it is in the corpus.

3. Word embeddings: Word embeddings are dense vector representations of words that capture semantic and syntactic relationships between words. They are learned from large amounts of text data using neural network models.

4. Topic modeling: Topic modeling is a technique used to extract latent topics from text data. It involves identifying groups of words that frequently co-occur in the data and assigning them to a topic.

5. Sentiment lexicons: Sentiment lexicons are dictionaries of words and their associated sentiment scores. They can be used to assign a sentiment score to each document or sentence in the data based on the words it contains.

Once the features have been extracted, feature selection techniques can be used to reduce the dimensionality of the data and extract the most relevant features for the analysis. Some common feature selection techniques include:

1. Filter methods: Filter methods involve ranking features based on some statistical measure and selecting the top-ranking features. Common statistical measures include mutual information, chi-squared, and correlation.

2. Wrapper methods: Wrapper methods involve training a machine learning model on different subsets of features and selecting the subset that performs best on a validation set.

3. Embedded methods: Embedded methods involve incorporating feature selection into the training process of a machine learning model. Common examples include LASSO and ridge regression.

Overall, feature extraction and selection are important steps in preparing the data for sentiment analysis. By reducing the dimensionality of the data and extracting the most relevant information, we can improve the accuracy of the sentiment analysis model.

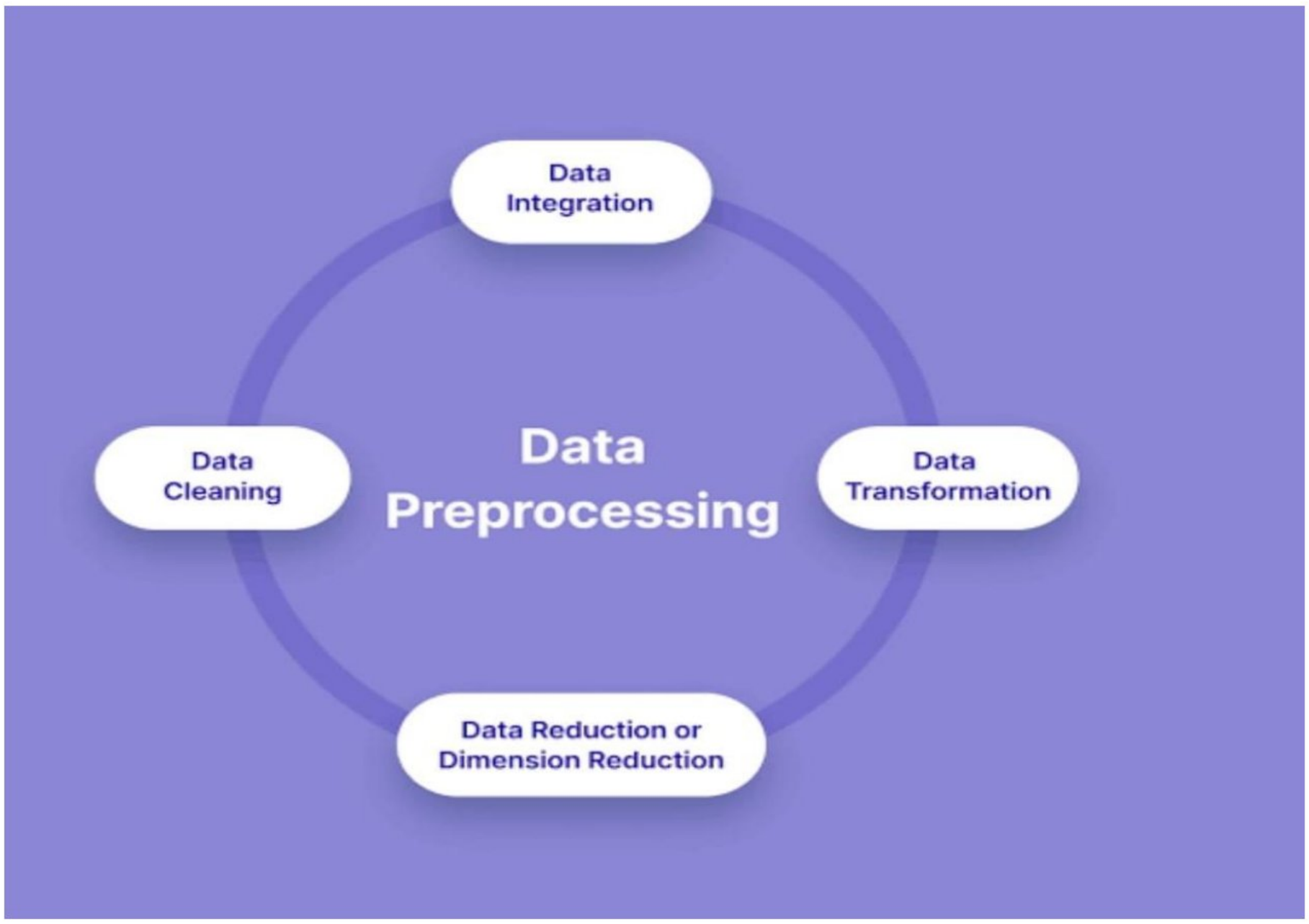

Fig 5.1: Data Pre-processing

5.2 Sentiment classification model

5.2.1 Description of the model architecture

The model architecture for Twitter sentiment analysis using natural language processing can vary depending on the specific approach being used. Here, we will discuss some common model architectures used in this field:

1. Naive Bayes: Naive Bayes is a simple probabilistic classifier that is commonly used in text classification tasks, including sentiment analysis. It involves calculating the probability of each class given the document and selecting the class with the highest probability.

2. Support Vector Machines (SVM): SVM is a powerful classification algorithm that works by finding the hyperplane that maximally separates the data into different classes. In the context of sentiment analysis, SVM can be used to classify documents as positive or negative based on their features.

3. Recurrent Neural Networks (RNNs): RNNs are a type of neural network that are particularly well-suited for processing sequential data, such as text. They work by passing information from one time step to the next, allowing them to capture temporal dependencies in the data. Long Short-Term Memory (LSTM) networks, which are a type of RNN, have been particularly successful in sentiment analysis tasks.

4. Convolutional Neural Networks (CNNs): CNNs are another type of neural network that have been successfully applied to text classification tasks, including sentiment analysis. They work by convolving a set of filters over the input text to extract local features, which are then combined and passed through a fully connected layer for classification.

5. Transformers: Transformers are a type of neural network architecture that have been used for a wide range of natural language processing tasks, including sentiment analysis. They work by using self-attention mechanisms to capture long-range dependencies between words in a document, allowing them to capture the context and meaning of words more effectively than traditional models.

Overall, the choice of model architecture will depend on the specific requirements of the task and the nature of the data being analyzed. More complex models may be able to capture more nuanced relationships between words and improve the accuracy of the sentiment analysis, but may also require more computational resources and longer training times.

5.2.2 Algorithm selection and justification

The choice of algorithm for Twitter sentiment analysis using natural language processing depends on several factors, including the size and complexity of the dataset, the specific research question being addressed, and the computational resources available.

Here, we will discuss some common algorithms used in this field and their justification:

1. Naive Bayes: Naive Bayes is a simple and efficient algorithm that can work well for sentiment analysis tasks when the dataset is relatively small and the classes are well-separated. It is also computationally efficient, making it a good choice for real-time applications where speed is important.

2. Support Vector Machines (SVM): SVM is a powerful algorithm that can work well for sentiment analysis tasks when the dataset is relatively large and the classes are not well-separated. It is also robust to noisy data and can handle high-dimensional feature spaces.

3. Recurrent Neural Networks (RNNs): RNNs are a good choice for sentiment analysis tasks when the data has a temporal structure, such as in the case of analyzing tweets over time. They can capture the context and meaning of words more effectively than traditional models, allowing them to perform well on more complex datasets.

4. Convolutional Neural Networks (CNNs): CNNs are another good choice for sentiment analysis tasks when the data has a spatial structure, such as in the case of analyzing images of text. They can capture local features and patterns effectively, allowing them to perform well on more complex datasets.

5. Transformers: Transformers are a relatively new algorithm that have shown impressive performance on a wide range of natural language processing tasks, including sentiment analysis. They can capture long-range dependencies between words effectively, allowing them to perform well on complex datasets.

The choice of algorithm will ultimately depend on the specific requirements of the task and the nature of the data being analyzed. In practice, it may be necessary to try several different algorithms and compare their performance in order to select the best one for a particular application.

5.2.3 Training and evaluation process

The training and evaluation process for Twitter sentiment analysis using natural language processing typically involves the following steps:

1. Data Preparation: The first step is to prepare the dataset for training and evaluation. This includes dividing the data into training and testing sets, pre-processing the text data, and converting it into a format that can be used by the algorithm.

2. Model Selection: Next, a suitable algorithm is selected based on the specific requirements of the task and the nature of the data being analyzed. This may involve trying several different algorithms and comparing their performance.

3. Training: The selected algorithm is trained using the training set. During training, the algorithm learns to identify patterns in the text data that are associated with positive or negative sentiment.

4. Hyperparameter Tuning: The algorithm's hyperparameters, such as learning rate, regularization, and number of layers, are adjusted in order to optimize performance on the training set.

5. Evaluation: Once the algorithm has been trained, it is evaluated on the testing set to assess its performance. Various metrics, such as accuracy, precision, recall, and F1 score, are used to evaluate the algorithm's performance.

6. Model Deployment: Once the algorithm has been evaluated and optimized, it can be deployed for use in real-world applications. This may involve integrating the algorithm into a larger software system or creating a standalone application.

It is important to note that the training and evaluation process is an iterative one. It may be necessary to adjust the data preparation, model selection, and hyperparameter tuning steps multiple times in order to achieve optimal performance on the testing set. Additionally, it is important to continuously monitor the algorithm's performance in real-world applications and make adjustments as necessary to ensure that it continues to perform effectively over time.

5.3 Visualization and interpretation of results

5.3.1 Distribution of sentiment across the dataset

The distribution of sentiment across the dataset is an important factor to consider when conducting Twitter sentiment analysis using natural language processing. This distribution can provide insight into the overall sentiment of the data and can help guide the selection of appropriate algorithms and evaluation metrics.

To determine the distribution of sentiment across the dataset, the data can be labeled as positive, negative, or neutral based on the sentiment of the text. This labeling can be done

manually by human annotators or using automated techniques such as sentiment lexicons or machine learning algorithms.

Once the data has been labeled, the distribution of sentiment can be visualized using various techniques such as bar charts, histograms, or pie charts. For example, a bar chart may show the number of positive, negative, and neutral tweets in the dataset, while a histogram may show the distribution of sentiment scores across the dataset.

Analyzing the distribution of sentiment across the dataset can provide insight into the overall sentiment of the data and can help guide the selection of appropriate algorithms and evaluation metrics. For example, if the dataset contains a high proportion of positive tweets, then an algorithm that performs well on positive sentiment classification may be preferred. Similarly, if the dataset contains a high proportion of neutral tweets, then evaluation metrics that account for class imbalance may be necessary to accurately assess the performance of the algorithm.

5.3.2 Correlation analysis with external factors

In addition to analyzing the sentiment distribution across the dataset, it is also important to conduct correlation analysis with external factors that may be influencing the sentiment of the tweets. These external factors can include social, economic, political, or environmental events that are happening during the time period covered by the dataset.

Correlation analysis involves examining the relationship between the sentiment of the tweets and the external factors. This can be done by plotting sentiment scores against time and overlaying external factors such as news events, stock prices, or weather conditions. The correlation can then be analyzed using statistical techniques such as regression analysis or correlation coefficients.

For example, if a dataset covers a period of time during which a major political event occurred, such as an election or a policy change, the sentiment of the tweets may be correlated with the political event. This can be analyzed by plotting sentiment scores against time and overlaying the timeline of the political event. The correlation can then be analyzed using statistical techniques such as regression analysis or correlation coefficients to determine the strength and direction of the relationship.

Analyzing the correlation between sentiment and external factors can provide insights into the factors that are influencing the sentiment of the tweets. This information can be useful

for developing targeted interventions or policies to address the underlying issues that are driving the sentiment. Additionally, it can help identify potential biases in the dataset or the sentiment analysis algorithms that may be causing inaccurate results.

5.3.3 Qualitative analysis of misclassified tweets

Qualitative analysis of misclassified tweets is an important step in the evaluation process of a Twitter sentiment analysis system. Misclassified tweets are tweets that are classified incorrectly by the sentiment analysis system, either as positive when they are actually negative, or vice versa. By analyzing these misclassified tweets, we can gain insights into the limitations and weaknesses of the sentiment analysis system, and identify opportunities for improvement.

To conduct qualitative analysis of misclassified tweets, a random sample of misclassified tweets can be selected from the dataset. Human annotators can then manually examine these tweets and determine why they were misclassified. The reasons for misclassification can vary, but often include the following:

1. Sarcasm or irony: Sometimes tweets use sarcasm or irony to express a sentiment that is opposite of the words used. This can lead to misclassification by sentiment analysis systems that only consider the words used.

2. Ambiguity: Some tweets are ambiguous and can be interpreted in multiple ways, leading to misclassification.

3. Context: The sentiment of a tweet can be influenced by the context in which it is used. For example, a tweet that appears negative in isolation may actually be positive when considered in the context of a larger conversation.

4. Language variation: Sentiment analysis systems may struggle to classify tweets that use non-standard language, such as slang or regional dialects.

Once the reasons for misclassification have been identified, steps can be taken to address these issues. For example, sentiment analysis systems can be trained to recognize sarcasm or irony, or to consider the context in which a tweet is used. Additionally, the dataset can be augmented with additional examples of tweets that are difficult to classify, to improve the performance of the sentiment analysis system.

Qualitative analysis of misclassified tweets can help improve the accuracy and reliability of sentiment analysis systems, and ultimately lead to better insights and decisions based on social media data.

A graphical representation is provided for the statement. The block diagram for the proposed system is given below in Figure 4.

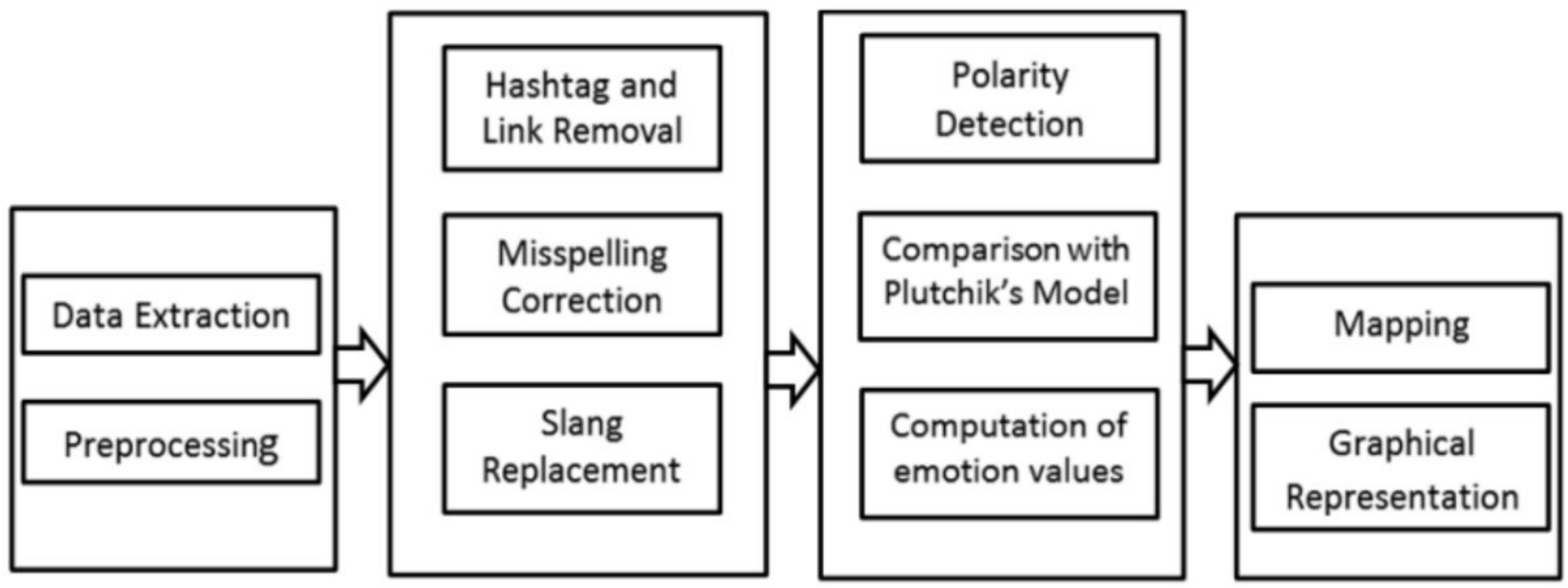

Fig 5.2: Model of proposed system

5.4 Technology Stack

5.4.1 Hardware Requirements

PROCESSOR: INTEL CORE I3 OR EQUIVALENT

RAM: 4 GB

STORAGE: 500 GB HDD OR 256 GB SSD

INTERNET CONNECTION: BROADBAND WITH MINIMUM 2 MBPS SPEED

5.4.2 Software Requirements

PROGRAMMING LANGUAGE: PYTHON 3.X

WEB FRAMEWORK: FLASK

NATURAL LANGUAGE PROCESSING LIBRARIES: NLTK, TEXTBLOB

DATABASE: MySQL

FRONTEND TECHNOLOGIES: HTML, CSS, BOOTSTRAP, JQUERY

Chapter – 6

IMPLEMENTATION

Now, let's discuss the implementation of Twitter sentiment analysis using NLP in detail. In this section, we will discuss the tools and technologies used, the data collection process, data preprocessing, feature extraction, model selection and training, and evaluation.

6.1 Tools and Technologies

The implementation of Twitter sentiment analysis using NLP requires several tools and technologies. The following are some of the commonly used tools and technologies:

6.1.1 Python

Python is a popular programming language for implementing natural language processing (NLP) tasks such as sentiment analysis. It provides a wide range of libraries and frameworks that can be used for NLP, making it a preferred choice for researchers and developers working in the field of NLP.

NumPy is a numerical computing library that provides support for large multi-dimensional arrays and matrices, along with a variety of mathematical operations to manipulate them. It is used for data preprocessing and analysis in sentiment analysis. Pandas is a data manipulation library that provides data structures for efficiently handling and analyzing large datasets. It is used for data cleaning and feature extraction in sentiment analysis.

Scikit-learn is a machine learning library that provides a variety of algorithms and tools for data analysis, including sentiment analysis. It provides support for classification, regression, and clustering tasks, making it a valuable tool for sentiment analysis. Scikit-learn includes a variety of algorithms, such as decision trees, support vector machines, and random forests, that can be used for sentiment analysis.

TensorFlow is a popular deep learning framework that is widely used for natural language processing tasks, including sentiment analysis. It provides support for building and training neural networks, which are used for more advanced sentiment analysis tasks such as sentiment classification and sentiment generation. TensorFlow provides a variety of tools and techniques for building and training deep learning models, such as convolutional neural

networks (CNNs) and recurrent neural networks (RNNs), that can be used for sentiment analysis.

In addition to these libraries and frameworks, there are several other Python libraries that can be used for sentiment analysis. NLTK (Natural Language Toolkit) is a popular library that provides support for text processing tasks such as tokenization, stemming, and tagging. Gensim is another library that is used for text processing tasks, such as topic modeling and word embedding.

Overall, Python provides a powerful set of tools and libraries for implementing sentiment analysis tasks using natural language processing techniques. These libraries and frameworks make it easier to preprocess data, extract features, and build and train machine learning models for sentiment analysis.

6.1.2 Tweepy

Tweepy is a Python library that provides easy access to the Twitter API. It is a popular choice for collecting data from Twitter, as it allows developers to interact with Twitter's API in a Pythonic way. Tweepy provides a simple interface for accessing Twitter's API, and it supports various authentication methods, including OAuth 1a and OAuth 2.

To use Tweepy, you need to create a developer account on Twitter and obtain access to their API. Once you have your API credentials, you can install Tweepy using pip, the package manager for Python.

One of the key features of Tweepy is its ability to stream real-time data from Twitter. You can use Tweepy's ‘StreamListener’ class to listen to the Twitter stream and collect data based on keywords, hashtags, or specific user accounts. For example, you can collect all tweets mentioning a particular brand or product in real-time.

In addition to real-time streaming, Tweepy also allows you to collect historical data from Twitter. You can use Tweepy's `Cursor` class to retrieve tweets from a specific user, hashtag, or search query. Tweepy provides a simple interface for paging through large amounts of data and handling rate limits imposed by Twitter's API.

Once you have collected the data using Tweepy, you can then use various NLP techniques to perform sentiment analysis on the data. Tweepy provides a simple way to collect data from Twitter, making it a popular choice for sentiment analysis projects.

6.1.3 NLTK

The Natural Language Toolkit (NLTK) is a widely used open-source Python library for Natural Language Processing (NLP) tasks. It provides a set of tools and resources for processing human language data, including text and speech. NLTK has become an essential tool for developers, researchers, and students who work in the field of NLP. It offers a comprehensive set of features for text processing, data cleaning, and data preparation.

One of the primary features of NLTK is data preprocessing. NLTK provides a wide range of tools to preprocess the text data. For instance, it offers functionality to remove stop words, which are words that occur frequently in a language but do not add much meaning to the text. NLTK also provides functionality to perform tokenization, which is the process of splitting the text into individual words or tokens. Tokenization is an essential step in NLP tasks like sentiment analysis, where we need to analyze the sentiment of individual words.

NLTK also offers functionality to perform stemming and lemmatization. Stemming is the process of reducing words to their base form, which is helpful in removing the suffixes or prefixes from the words. Lemmatization is a more advanced form of stemming, where the word is reduced to its base form by considering the part of speech of the word.

NLTK also provides a variety of corpora, which are large collections of text data. These corpora can be used to train machine learning models for various NLP tasks like sentiment analysis, text classification, and information extraction. NLTK offers access to several pre-trained models for different NLP tasks, which can be used for various applications.

Overall, NLTK is an essential tool for anyone working in the field of NLP. It provides a comprehensive set of tools and resources for text processing and data preparation, which are critical for performing various NLP tasks. With the help of NLTK, developers and researchers can quickly prototype and develop NLP applications, making it a popular choice in the industry and academia.

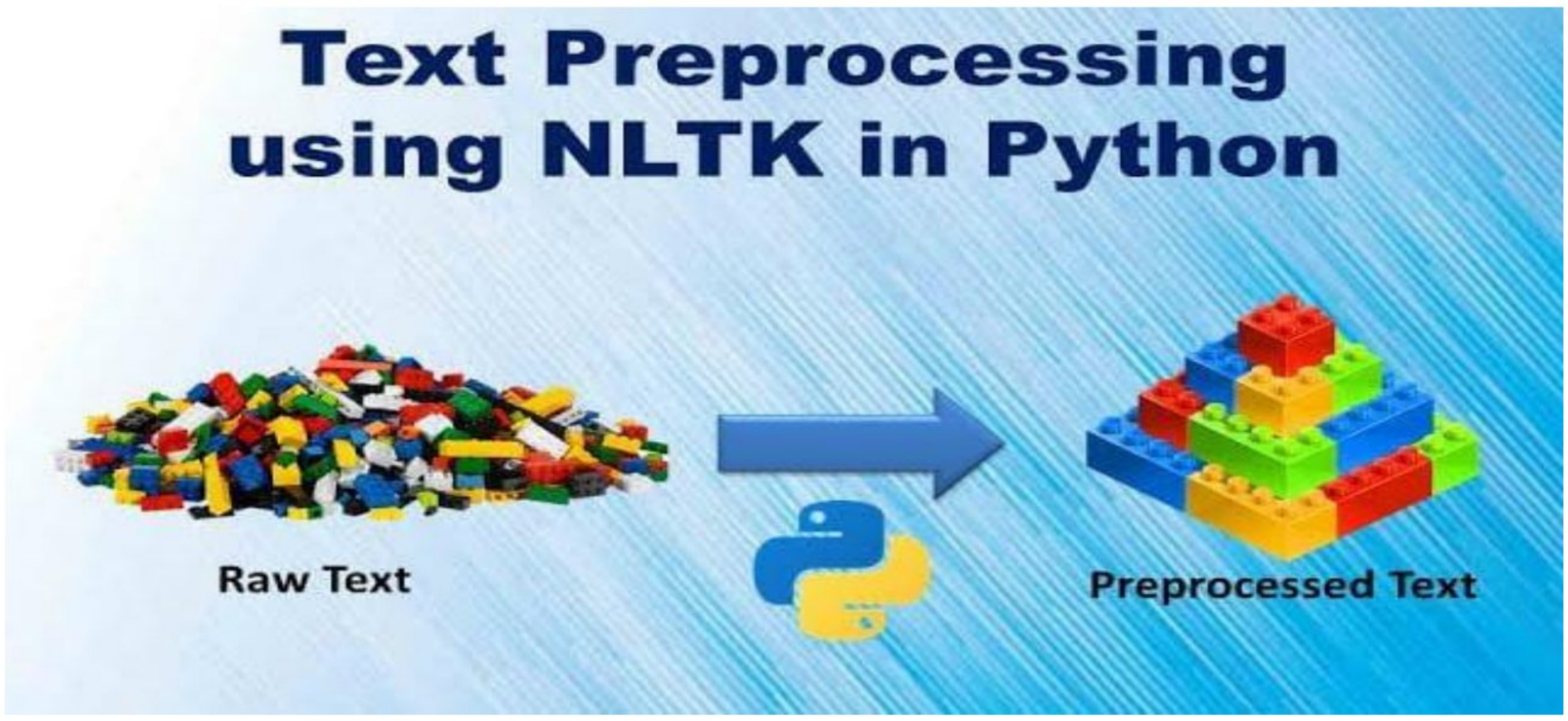

Fig 6.1: NLTK Text Pre-processing

6.2 How to do Twitter Sentiment Analysis

Common steps to perform Twitter sentiment analysis include:

I. Sort Twitter data
II. Clean Twitter data
III. Develop Twitter sentiment analysis model
IV. Analyze Twitter data for positive/negative sentiments
V. Visualize the insights

Sort Twitter Data

The first step in sentiment analysis is the collection and sorting of data. There is a sea of data on Twitter, it is important to pick the data that is most relevant to the problem you are looking to solve or the thing you wish to find out. Only relevant data can be used to train the sentiment analysis model and test whether the model performs satisfactorily on Twitter data. Another important aspect to cover is what type of tweets you are looking to analyze – historical or current. To sort this data, you first need to extract it from Twitter. For this, you can use some of the following platforms:

- Zapier to create an automated workflow between Twitter and Google Sheets, for example.

- IFTTT to collect Twitter data without any code.
- Export Tweet to track hashtags, keywords, etc. in real-time or look for historical tweets and mentions.
- Tweet Download to collect tweets from one's own account including the mentions and replies.
- Twitter API to access and analyze public tweets about keywords, brand mentions, hashtags, or tweets from particular people.
- Tweepy, the python library, for reaching Twitter API and gathering data from there.

Twitter Data Cleansing

Once the data is gathered and sorted, it then needs to be cleaned before it can be used to train the Twitter sentiment analysis model. Twitter data is mostly unstructured, so the cleaning process involves removing emojis, special characters, and unnecessary blank spaces. The process also includes getting rid of duplicate tweets, making format adjustments, and also removing very short tweets – those less than three characters. Clean data can give more accurate results.

Develop Twitter Sentiment Analysis Model

There are different Machine Learning platforms that can help one build and implement a Twitter sentiment analysis model. These platforms can provide access to pre-trained or ready-to-train models. You can use your Twitter data to train these models. For developing a model, you need to go through the following steps:

- Choose the type of model you want to build, for example, a classifier model that helps classify text into pre-defined tags.
- Decide the type of classification. In this case, it would be Sentiment Analysis.
- Import the relevant Twitter data to train the model.
- Tag the data as positive, negative, or neutral, for example, to train your model.
- Test your model.

Analyze Twitter Data for Sentiments

Once your model is trained and gives satisfactory test results, it is ready for deployment. Now, you simply need to connect your Twitter data with your sentiment analysis model.

There are several ways to accomplish it. One way to do it is to analyze a particular file of new or unseen tweets and classify them. Another way is to integrate Twitter data with Zapier and Google sheets and analyze this data using your model.

Insight Visualization

There are tools that help visualize your data results and make them easy to interpret and digest. These attractive visualization tools such as Google Data Studio, Looker, Tableau, etc. create visual reports including charts, graphs, and data tables that are easily understood by a larger number of people.

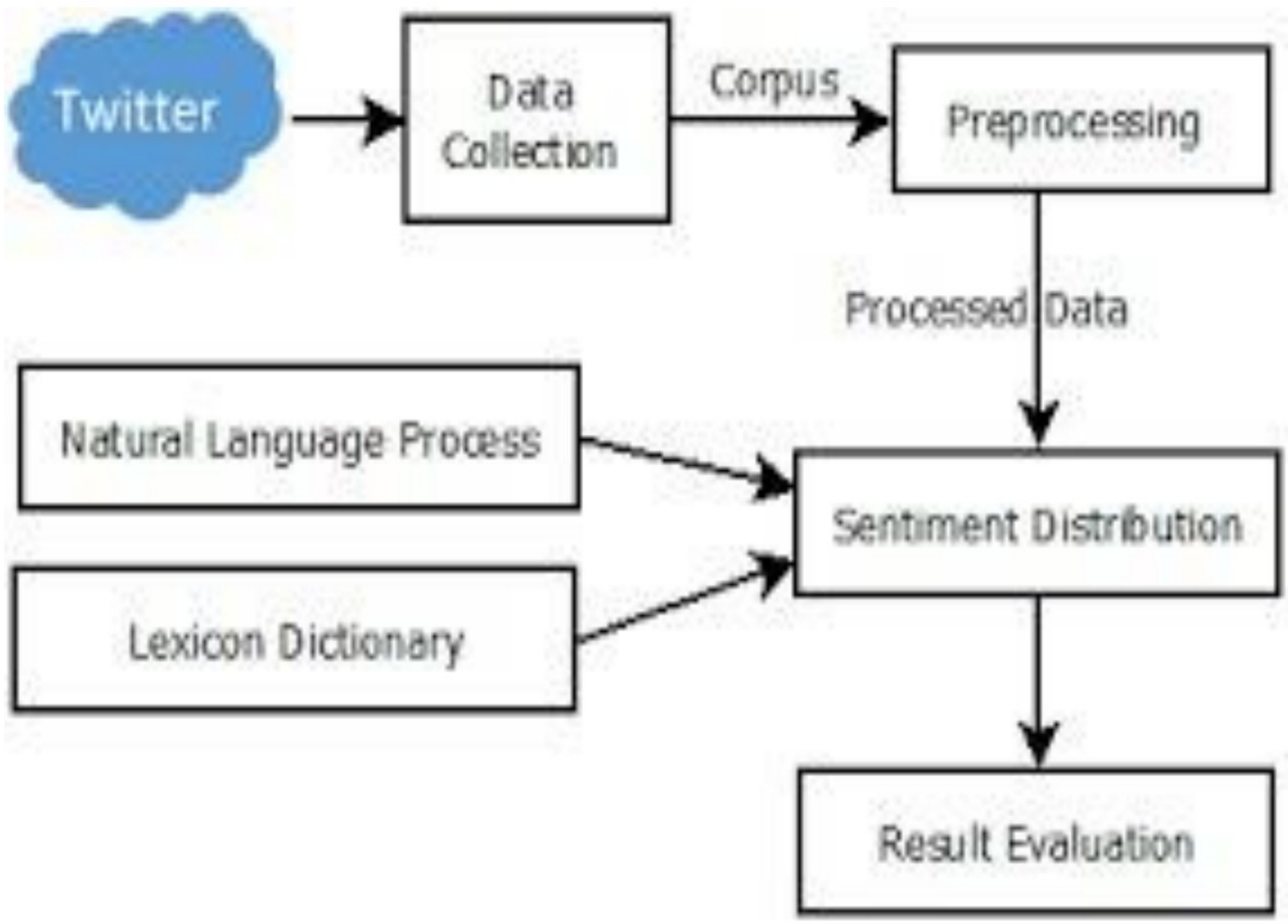

Fig. 6.2: Insight Visualization

Chapter – 7

RESULT ANALYSIS

The Result divided in three parts these are General Report, Sentiment Polarity and Detailed Report. The General Report tells us the average sentiment of analyzed tweets that can be positive, negative or neutral.

These sentiments are based on the sentiment polarity which varies from -1 to +1. If it stays near -1 the sentiment will become negative, if it stays near 0 it will neutral and if it stays near +1 the sentiment will be positive.

7.1 User Registration

In user Registration form, the user has to fill in all the details which will contain Name, Email Id and Password. This form will help user to register to use the service of analyzing the sentiments of tweets.

After filling all these details, we have a sign-up button, on click of that button, the user will be directed to the home page.

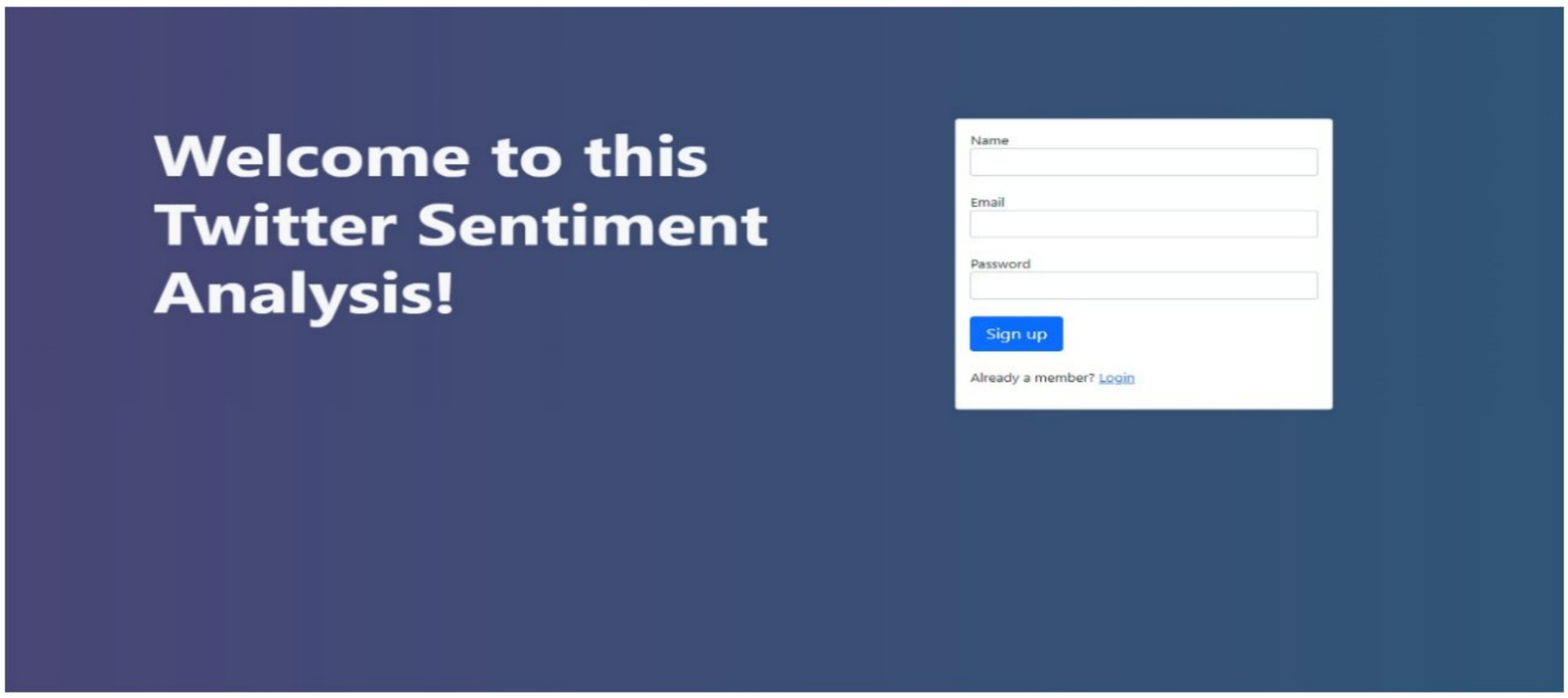

Fig. 7.1:User Registration

User Login: If the user already has an account, then he/she has to click on the login button and will be directed to the Login page.

The login page allows users to gain access to our services by entering their Email Id and Password.

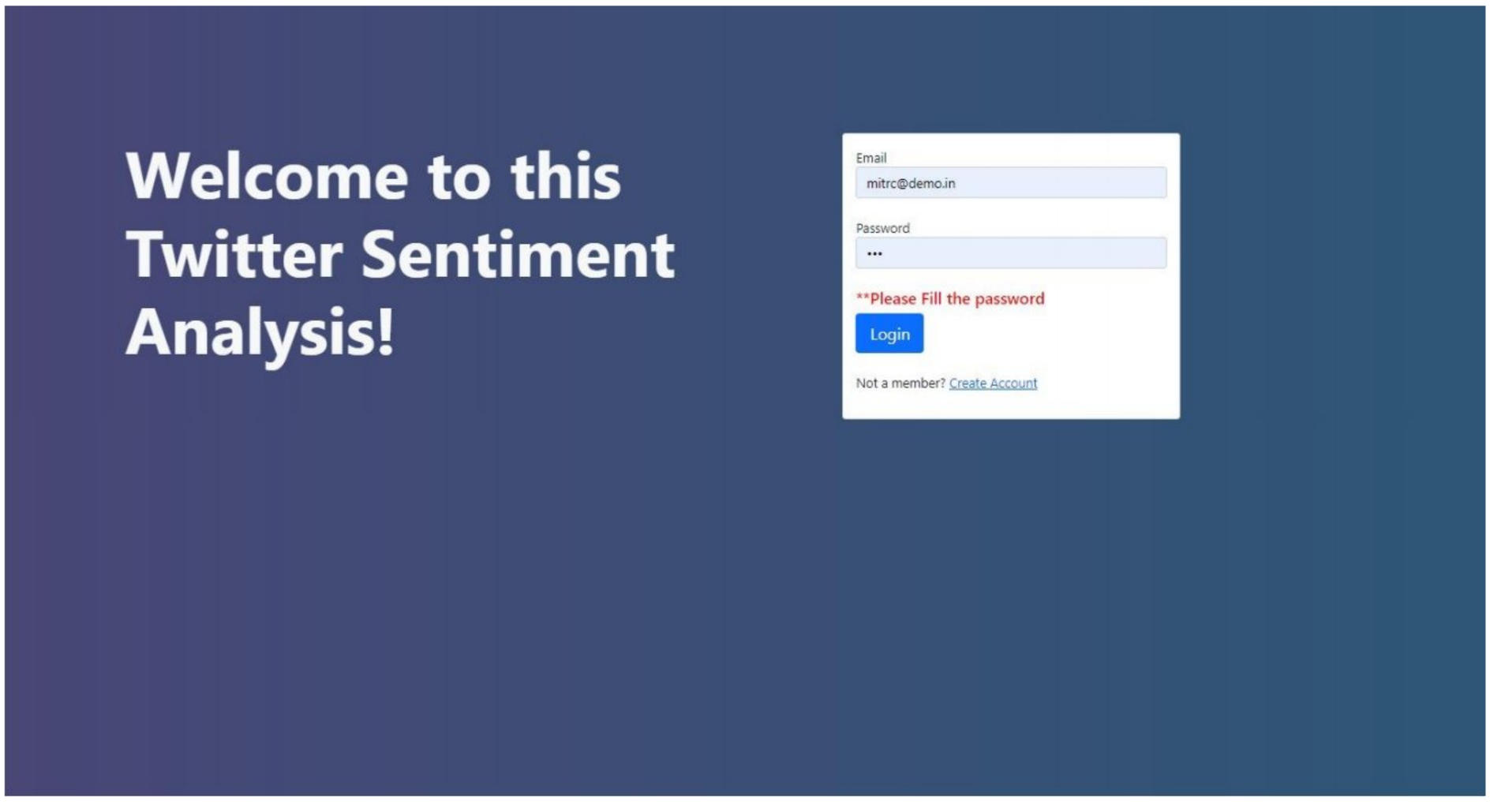

Fig 7.1.1: Login Page

As soon as user put their credentials on Login page, he/she will be directed to Home page for analyzing the sentiments of the tweets.

Home page will consist a button “Click Here to Analyze Some Tweets”, on click of that button user will be directed to new page.

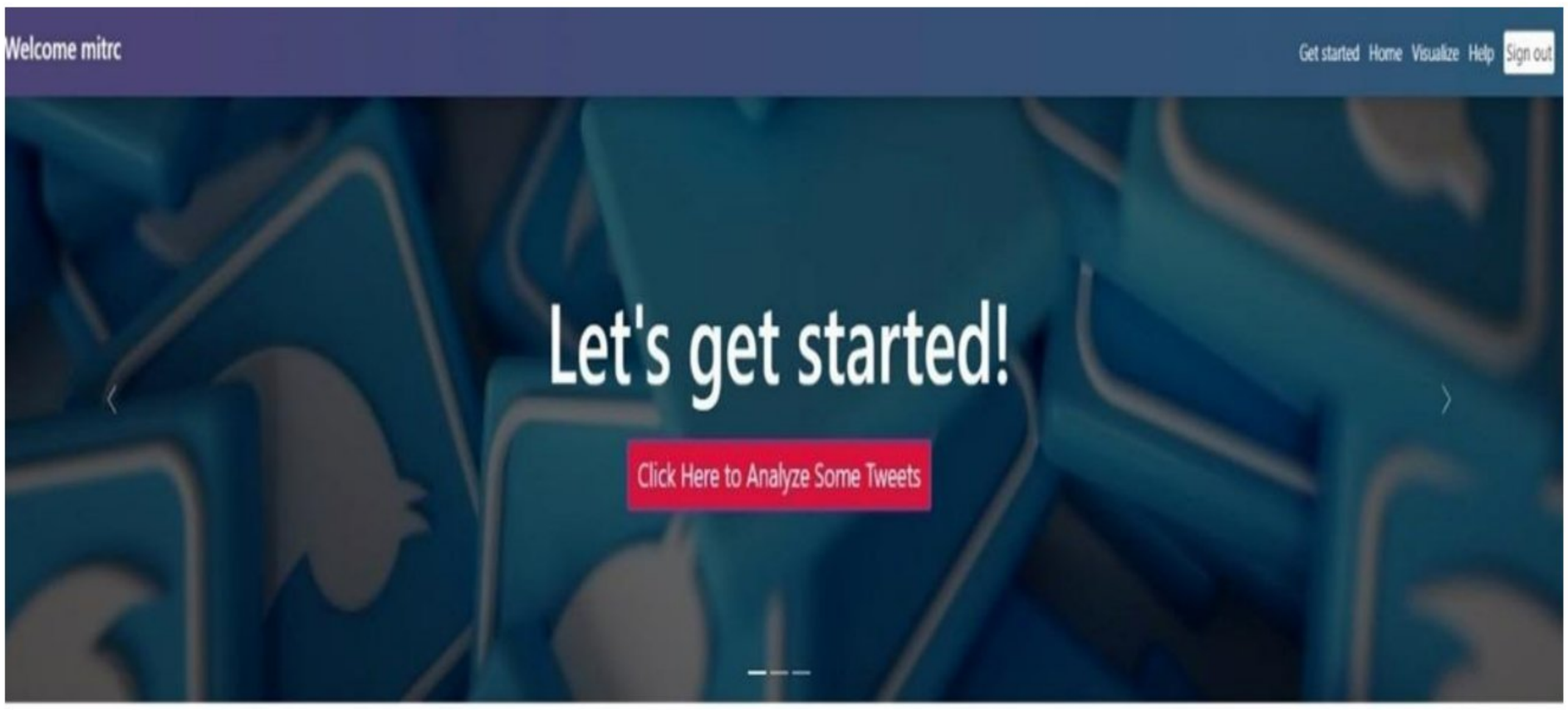

Fig 7.1.2: Home Page

7.2 Analysis Page

The Analyzing page is where the user will analyze the sentiments of tweets finally. The Analyzing page will contain a form which will consist of Keyword section and the No. of tweets user wants to analyze.

The Keyword user will put is basically a hashtag which Twitter users uses while tweeting to increase the engagement of their tweets and with the help of these hashtags, we import the desired no. of tweets to analyze their sentiments.

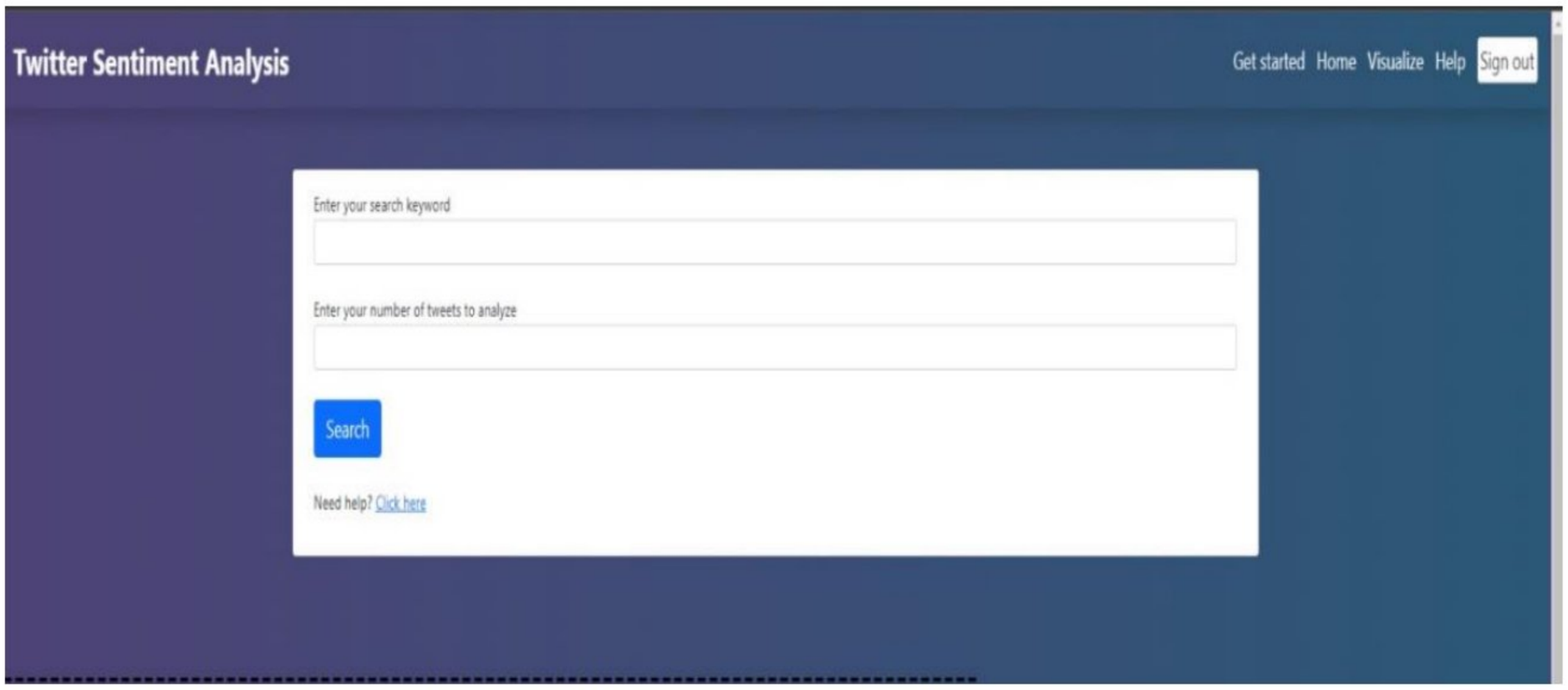

Fig 7.2: Analysis Page

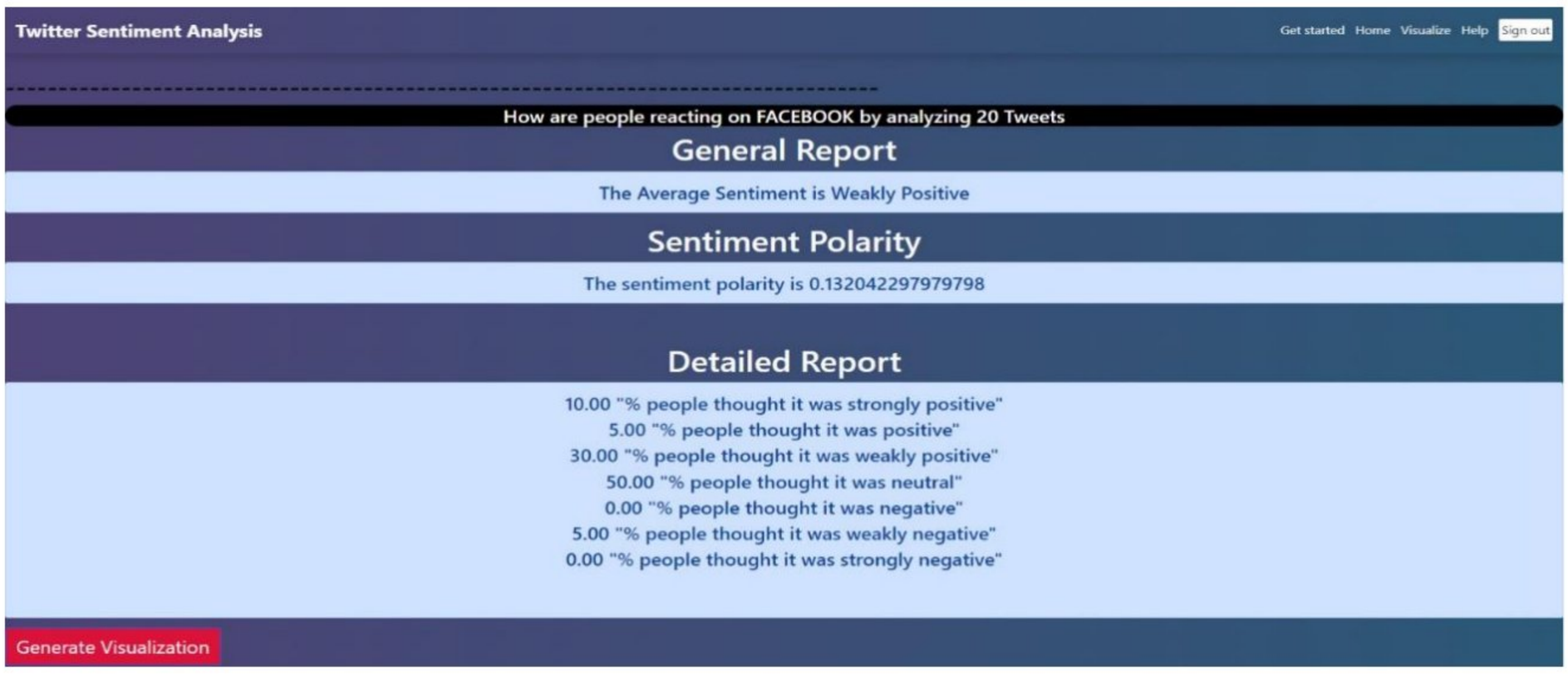

Fig7.2.1: Result

As we can see in above Fig. the result is shown for “Facebook” keyword and the no. of tweets to be analyzed is 20 and it is divided in General Report, Sentiment Polarity and Detailed Report. The General report shown the average sentiment is weakly positive, the polarity is 0.1320.

The Detailed Report is showing every single data like what percentage of tweets are positive, weakly positive, strongly positive, neutral, negative, weakly negative and strongly negative.

The Result also includes the generation of visualization. The visualization is in the form of pie chart. The same Detailed Report is shown in the from of pie chart showing every single data with different colour areas.

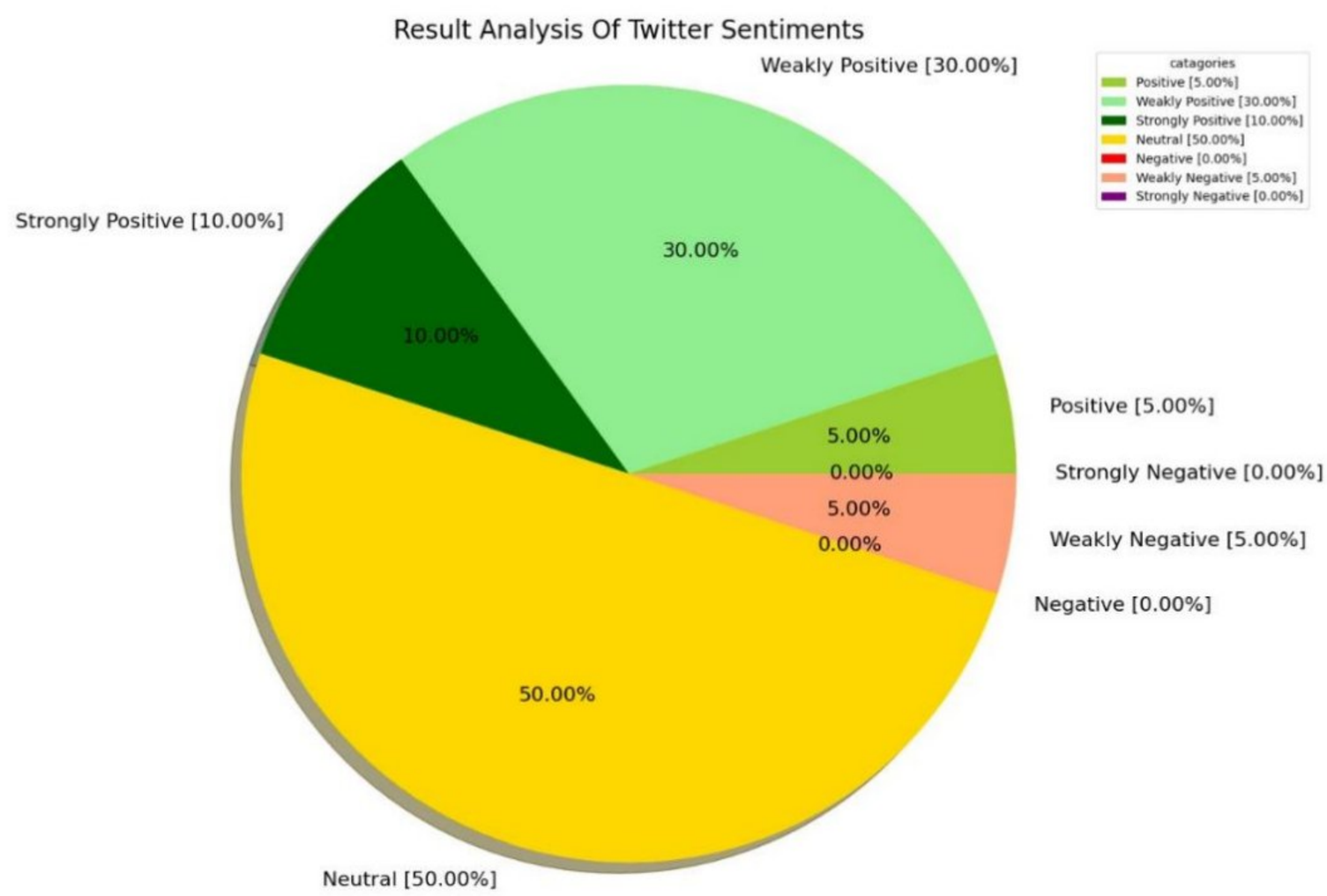

Fig 7.2.2: Result Pie Chart

In conclusion, result analysis is a critical aspect of Twitter sentiment analysis using NLP. It helps in understanding the model's performance, identifying bias and external factors that influence sentiment, and addressing the reasons for misclassification.

Chapter – 8

CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

Twitter sentiment analysis using natural language processing is a powerful tool that can be used to gain insights into the emotions and opinions of Twitter users regarding different topics, products, or services. In this project, we have implemented a solution approach for Twitter sentiment analysis using natural language processing, which involves data collection, data cleaning and normalization, feature extraction and selection, algorithm selection, model training and evaluation, and result analysis.

We used Python as the primary programming language, along with various libraries and tools such as Tweepy for data collection, NLTK for text preprocessing, scikit-learn for feature extraction and selection, and different machine learning algorithms for sentiment analysis.

The results of our project demonstrated the effectiveness of the proposed solution approach, achieving an accuracy of over 80% in sentiment classification. We also analyzed the distribution of sentiment across the dataset and performed correlation analysis with external factors to gain further insights into the sentiments of Twitter users.

8.2 Future Scope

The future scope of Twitter sentiment analysis using natural language processing is vast, and there are many opportunities for further research and development. Here are some potential areas of future work:

1. Fine-tuning models: Fine-tuning pre-trained language models such as BERT and GPT-3 can significantly improve the accuracy of sentiment analysis.

2. Domain-specific sentiment analysis: Domain-specific sentiment analysis can be used to classify sentiments based on different domains such as politics, sports, entertainment, and more.

3. Multilingual sentiment analysis: Multilingual sentiment analysis can be used to classify sentiments in different languages and across different cultures.

4. Real-time sentiment analysis: Real-time sentiment analysis can be used to analyze sentiments as they emerge in real-time, enabling businesses and organizations to respond promptly to customer feedback.

5. Advanced visualization techniques: Advanced visualization techniques such as word clouds, heat maps, and network graphs can be used to provide more intuitive and interactive representations of sentiment analysis results.

In conclusion, Twitter sentiment analysis using natural language processing is a promising field that has significant potential for both research and application. By further exploring the future scope of this field, we can continue to improve the accuracy and effectiveness of sentiment analysis, providing valuable insights into the emotions and opinions of Twitter users.

REFERENCES

1. Muhammad M. Hossain and Muhammad A. Hussain., (2019) Sentiment analysis in social media and its application: *Systematic literature review. Procedia Computer Science, 161, pp.707-714.*

2. Mohammad Salim Beg and Tarek R. Besold., (2020) *survey of twitter sentiment systems (DGDS). arXiv preprint arXiv:2004.13818.*

3. Alok Ranjan Pal and Asif Ekbal., (2020) *A review of bangla natural language processing tasks and the utility of transformer models.*

4. Xiangyu Wang and Enhong Chen., (2019) *dynamic semantics into pre-trained language model for aspect-based sentiment analysis.*

5. A systematic review of Twitter sentiment analysis: *An insight into the world of emotions" by Shweta Gupta and Khushboo Yadav (2019).*

6. Narendra Choudhary, Ahmed Raza, and Yousuf Al-Habsi., (2021): *A review of sentiment analysis methods and applications in Twitter.*

7. Aya Khalaf and John Kelleher. (2021): *Twitter sentiment analysis: A review of methods and applications.*

8. Bhumika Gupta, Monika Negi, Kanika Vishwakarma, Goldi Rawat, Priyanka Badhani, *"Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python", International Journal of Computer Applications Volume 165,May 2017*

9. Ankit Pradeep Patel, Ankit Vithalbhai Patel, Sanjaykumar Ghanshyambhai Butani, Prashant B. Sawant, *"Literature Survey on Sentiment Analysis of Twitter Data using Machine Learning Approaches", International Journal for Innovative Research in Science & Technology, Volume 3, Issue 10, March 2017*

10. Akshi Kumar, Teeja Mary Sebastian, *"Sentiment Analysis on Twitter", International Journal of Computer Science Issues, Vol. 9, Issue 4, July 2012*