
This is the **published version** of the master thesis:

Kulkarni, Saurabh Shrikant; Serra-Sagristà, Joan, dir. An overview of Sentiment Analysis of Twitter Data. 2020. 70 pag. (1170 Màster Universitari en Enginyeria de Telecomunicació / Telecommunication Engineering)

This version is available at <https://ddd.uab.cat/record/259528>

under the terms of the  license



**Universitat Autònoma
de Barcelona**

A Thesis for the

Master in Telecommunication Engineering

An overview of Sentiment Analysis of Twitter
Data

by
Saurabh Shrikant Kulkarni

Supervisor: Joan Serra Sagrista

Department of Information and Communication Engineering

**Escola d'Enginyeria (EE)
Universitat Autònoma de Barcelona (UAB), Bellaterra**

September 2020



El sotasignant, *Joan Serra Sagrista*, Professor de l'Escola d'Enginyeria (EE) de la Universitat Autònoma de Barcelona (UAB),

CERTIFICA:

Que el projecte presentat en aquesta memòria de Treball Final de Master ha estat realitzat sota la seva direcció per l'alumne *Saurabh Shrikant Kulkarni*.

I, perquè consti a tots els efectes, signa el present certificat.

Bellaterra, September 2020.

Signatura: *Joan Serra Sagrista*

Abstract

In the last few years, social media has seen tremendous growth in the number of users. In particular, Twitter has revealed to be one of the most widespread microblogging services for instantly publishing and sharing opinions, feedbacks, ratings, etc., contributing to the development of the emerging role of users as sensors. Twitter has become the largest source of obtaining data worldwide. This project proposes a method to predict the future of the entertainment industry, telecommunication industry, and other various industries. However, due to the huge amount of data to be collected and analyzed and limitations on data access imposed by Twitter public APIs, more efficient requirements are needed for analytics tools, both in terms of data ingestion and processing, as well as for the computation of analysis metrics, to be provided for deeper statistic insights and further investigations.

This project evaluates people's feelings about different products related to various industries. Twitter API is used to access the tweets directly from Twitter and form a model for sentiment classification. The result of the analysis is characterized by positive, negative, and neutral observation from the user's opinions.

Acknowledgements

First, I would like to thank my thesis supervisor, Prof. Joan Serra Sagrista for invaluable advice and input. I have regarded our regular meetings as very enjoyable and beneficial to the quality of my work. Thank you for mentoring in every possible way throughout the thesis.

Second, I would like to thank all my professors who guided and helped me throughout the masters program at Autonomous University of Barcelona.

Third, I would like to thank my colleagues Alejandro Perez, Victor Pinero, and Xavier Colin for maintaining camaraderie in a master's degree.

Last but not least, I would like to thank my family for being a pivotal column through life. Your support has provided me the strength to achieve anything in life.

Contents

Abstract	i
Acknowledgements	iii
1 Introduction	1
1.1 Big Data Analysis	1
1.2 Social Media - A perspective	3
1.3 Problem Statement	4
1.4 Research Objective	4
1.5 Goal/ Rationale	5
1.6 Document Structure	5
2 Background and Related Work	7
2.1 Introduction	7
2.2 Literature Review	7
2.3 Software Requirements	9
2.4 Parameters for models	9
2.5 Sharing Work and Knowledge	9
3 Experimental Design	10
3.1 Introduction	10
3.2 Goal	10
3.3 Twitter	12
3.4 Twitter API	13

3.5	Proposed Design	14
3.6	Twitter Authorization	17
3.7	Data Collection	18
3.8	Data Preprocessing	18
3.9	Classifiers	19
3.9.1	Naive - Bayes Classifier	20
3.9.2	Bag of Words Classifier	21
3.10	Conclusion	22
4	Execution	23
4.1	Introduction	23
4.2	Software Installation	23
4.3	Tokenization	24
4.4	Preparation	25
4.4.1	Tweepy	26
4.4.2	Textblob	26
4.5	Data Preprocessing	27
4.6	Emoticons	29
4.7	Framework	29
4.8	Model Execution	30
4.9	Use of Textblob	33
4.9.1	Polarity	34
4.9.2	Subjectivity	34
4.10	Google Translator API	36
5	Analysis	37
5.1	Introduction	37
5.2	Technological Products	37
5.3	Analysis of Web Series and TV series	43
5.4	Movies	48

5.5	Conclusion	50
6	Conclusions and Future Work	51
6.1	Conclusions	51
6.2	Future Work	52
	Bibliography	52

List of Tables

4.1	List of substituted emoticons	29
5.1	Cisco	37
5.2	Huawei Products	41
5.3	Breaking Bad	44
5.4	UPLOAD Web Series	47
5.5	Movies Database	49
5.6	Reviews for Avengers: Endgame	50

List of Figures

2.1	Training a neural network using Classifier algorithms	8
3.1	Experiment Activities	14
3.2	API interface with authorization	17
3.3	Sentiment Classification Algorithms[5]	19
3.4	Bayes Theorem	20
3.5	Bag of Words	21
4.1	Tweepy Implementation	26
4.2	Tweet Example	27
4.3	Framework of sentiment analysis[7]	28
4.4	Tweet about Apple	30
4.5	Example showing tweet and feature words	32
4.6	Sample Tweet about 'Bard of Blood'	34
4.7	Sentiment Function Results	35
4.8	Text translation JSON	36
5.1	Visualization of Tweets about Cisco	38
5.2	Visualization of Tweets about Cisco based on precision matrix	39
5.3	Pie Chart representing Sentiment Analysis of Huawei P30 pro	41
5.4	Horizontal Bar graph for Sentiment Analysis of Samsung S10	42
5.5	Tweet about Breaking Bad	44
5.6	Breaking Bad	45
5.7	Money Heist	46

5.8	UPLOAD	47
5.9	UPLOAD review for each day	48
5.10	Sample Tweet About Lagaan	49
5.11	Sentiment Analysis of Indian Movies	50

Chapter 1

Introduction

The current scenario in analyzing the data is tedious and takes a lot of time. Data streams are dynamic and constantly upgrading. The different techniques need to be compared with each other to identify the better technique amongst the current methods. Sentiment analysis (a.k.a opinion mining) is the automated process of identifying and extracting the subjective information that underlies a text. This can be either an opinion, a judgment, or a feeling about a particular topic or subject. The most common type of sentiment analysis is called ‘polarity detection’ and involves classifying a statement as ‘positive’, ‘negative’, or ‘neutral’.

This master thesis reports on the Sentiment Analysis of data generated from social media websites, mainly focused on Twitter using various algorithms followed by the implementation of the algorithms.

1.1 Big Data Analysis

Big data is a term that defines data set are so large or complex that traditional data processing applications are inadequate[3]. The need for big data doesn’t revolve around what amount of data you have, but what you want to do with it. Big Data is a term used to describe a collection of data that is huge in volume and yet growing exponentially with time. In short such data is so large and complex that none of the traditional data

management tools can store it or process it efficiently.

Analysis of big data allows analysts, researchers and business users to make better and faster decisions using data that was previously inaccessible or unusable. Businesses can use advanced analytics techniques such as text analytics, machine learning, predictive analytics, data mining, statistics and natural language processing to gain new insights from previously untapped data sources independently or together with existing enterprise data.

The most common example of big data is social networking websites like Facebook, Instagram, Twitter. According to statistics, 500+ Terabytes of data are ingested into the databases social media websites like Facebook and Twitter. The data is generated by the means of uploading photos, message exchanges, and writing comments on social media websites.

1.2 Social Media - A perspective

The social media websites like Facebook, Instagram, Twitter are the modern sources to express opinions, to communicate with people, and to share thoughts. Amongst these, Twitter is the most suitable social networking websites that convey one's thoughts and establish a connection between the users with minimum use of words.

Large chunks of data are engendered with these social media websites. The data is disorganized and needs to be monitored and structured for analyzing data patterns. These data sets are used for both research and commercial purposes. Social media analysis is an emerging interdisciplinary research topic which with the effort of combining, extending and adapting techniques and methods for data analysis, monitoring, and visualization.

1.3 Problem Statement

The current scenario in data analysis twitter data is tedious and takes a lot of time. Data streams are dynamic and constantly upgrading. The different techniques need to be compared with each other to identify a better technique amongst the current methods.

1.4 Research Objective

The research objective should formulate the solution to the problem with the use of sentiment analysis techniques for the data analysis. Sentiment analysis (a.k.a opinion mining) is the automated process of identifying and extracting the subjective information that underlies a text. This can be either an opinion, a judgment, or a feeling about a particular topic or subject. The most common type of sentiment analysis is called ‘polarity detection’ and involves classifying a statement as ‘positive’, ‘negative’, or ‘neutral’.

Sentiment Analysis aims towards determining the point of view of a speaker or writer towards any topic or incident. Sentiment analysis is broadly classified in the two types first one is a feature or aspect-based sentiment analysis and the other is objectivity based sentiment analysis. Sentiment analysis of Twitter is complicated as compare to broad-ranging sentiment analysis as it has many slang words, spelling mistakes, and repeating characters or words.

1.5 Goal/ Rationale

The final goal of the study is to compare the various techniques used for Sentiment analysis using algorithms developed. The comparison will be based on accuracy, classification algorithms, and a few examples based on the use of algorithms. The examples include product reviews(Technological products), a wide range of movies, and TV/Web series reviews posted by users on Twitter.

1.6 Document Structure

This document aims to offer a research study in a meticulous structure. Such a structure makes it easier to pinpoint relevant information and lowers the risk of missing any information. The study is presented as follows:

Chapter 2: Background and Related Work clarifies how this study is related to the existing work (literature review) including the algorithms developed, software requirements specifications, type of models used, and accuracy rates of the different models.

Chapter 3: Experimental Design describes the outcome of the experiment planning phase, including goals, hypotheses, parameters, variables, design, participants, objects, instrumentation, data collection procedure, analysis procedure, and evaluation of the validity.

Chapter 4: Execution describes each step in the production of the experiment, including the sample, preparation, data collection and actions performed on the data.

Chapter 5: Analysis compiles the sentiment analysis of the data collected via Twitter, treatment of the data and display statistical readings based on analysis. It presents the evaluation of the results, limitations of the study and learning outcomes from the research.

Chapter 6: Conclusions and Future Work presents the summary of the study including impacts and future work.

A list of references is mentioned afterwords.

Chapter 2

Background and Related Work

2.1 Introduction

This chapter describes the existing work relevant to the topic of sentiment analysis by various authors. It includes algorithms used to classify the data, results obtained from the experimental extraction of the data.

2.2 Literature Review

Social media analysis is an emerging research topic with the effort of combining, extending, and adapting techniques and methods for social media data analysis, monitoring, and visualization.[4] Various techniques were developed to perform data analysis of social media websites. A lot of work has been carried out in the field of sentiment analysis for the live data from the users to extract the sentiments of common people towards any topic, trend, products, etc. The studies mainly focus on extracting useful information from the natural language of users and process it to get the real sentiments[1].

Lei Wang and John Q Gan [2] proposed a method to use a candidate's popularity prediction based on the Twitter data analysis to predict the results of the French election results 2017. The author used a technique that identifies the tone of the user and differentiates the tweets accordingly.

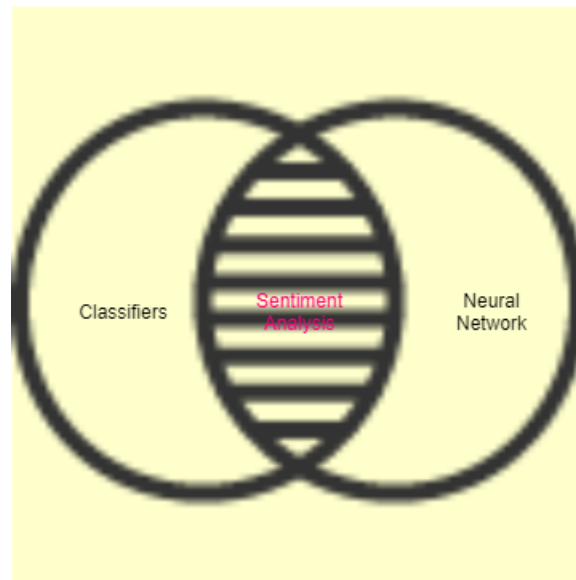


Figure 2.1: Training a neural network using Classifier algorithms

Another author used software called Rapid-miner[1] to perform sentiment classification based on text mining and data mining. The author used two different classifiers for the data classification that are Naïve Bayes and K-NN.

As sentiment analysis is a sub-field of NLP(Natural language processing), the author suggested[3] the formation of a twitter vigilance network that can act as a multipurpose machine to collect twitter data and analyze the data from various language tweets.

Long before the data collection, preliminary experimentations were done using classifiers like Artificial Neural Network, Naïve Bayes Classifier, and SVM.[8] The data needs to be filtered as emojis are not identified by the ASCII. The author suggested the use of the Textblob library available in Python to train a classifier[8].

2.3 Software Requirements

The neural network is created using classifiers for training purposes. The programming language used is Python or R. The existing libraries available are installed for classifying the data or text extracted.

2.4 Parameters for models

The accuracy and validity of models created for numerous algorithms are achieved using metrics like measurements and cross-validation.[6] The accuracy of the models is achieved with the help of comparing the existing datasets containing a list of words with a positive and negative connotation and frequency of the feature words used in a text.

Accuracy: A measure of how often a sentiment rating was correct. For documents with tonality, accuracy tracks how many of those that were rated to have tonality were rated correctly. While building the system, following measure is considered in the final model:
Accuracy : $[\text{Num. of Correct Queries} / \text{Total Num. of Queries}]$

2.5 Sharing Work and Knowledge

It is important to build a dialogue between tacit and explicit knowledge. A lack thereof can lead to a superficial interpretation of existing knowledge that has little to do with reality may fail to embody knowledge in a form that is concrete enough to facilitate further knowledge creation or has little shareability.

Chapter 3

Experimental Design

3.1 Introduction

This chapter presents the procedures followed for the experimental research. This served as a blueprint of the execution and analysis phase.

The design is based on the research goal and hypotheses that support it. A matching research design is then selected. Following that, the details of the experimental design are discussed, including its parameters, variables, planning, expected participants, objects, instrumentation and procedures for data collection and analysis. Finally, the validity of the experimental design is evaluated.

3.2 Goal

When evaluating the sentiment (positive, negative, neutral) of a given text document, research shows that human analysts tend to agree around 80-85% of the time. This is the baseline we (usually) try to meet or beat when we're training a sentiment scoring system. But this does mean that you'll always find some text documents that even two humans can't agree on, even with their wealth of experience and knowledge. The accuracy intended to achieve from the experiment should be matched with the human analyst.

But when you're running automated sentiment analysis through natural language processing, you want to be certain that the results are reliable. So, how accurate can we get, and how can we ensure the best-possible sentiment accuracy? After the classification of the text according to three parameters termed as 'positive', 'negative', and 'neutral', the final results are to be presented in various histograms.

The metric precision/accuracy focuses on the implemented approach. It calculates the amount of correctly assigned posts in relation to all automatically classified posts.

$$Accuracy : \frac{TruePositive}{TruePositive + FalsePositive} \quad (3.1)$$

Precision/Accuracy measures the exactness of a classifier. A higher precision means less false positives, while a lower precision means more false positives.

3.3 Twitter

Twitter is a social media website that allows people to express their feelings, opinions, and interact with other people with the help of 'Tweets'. Due to the limited number of words, it is also referred to as a micro-blogging website. Worldwide around 300 million users are active daily on Twitter.

Twitter has emerged as one of the most widespread environment for social media analytics [4], with over 3 billion tweets and 15 billion API calls generated daily. Followers receive notifications connected to the actions performed by the users they follow. Typical actions of users can be: posting a message (tweet), commenting, expressing like/favorite, retweeting (the echo of some tweets by some users to the followers of the retweeting user). Therefore, tweets and retweets are exposed to other Twitter users, thus enhancing the chance of provoking their interests and reactions. Some of these mechanisms can generate viral processes that may lead to a huge diffusion of tweets in the user community[4].

3.4 Twitter API

Twitter developer labs have generated various API products for developers to analyze data generated on Twitter. Using Twitter's API, users can generate, search, and filter tweets. It can generate tweet timelines for the keyword. Using Twitter API, an application is created for a specific purpose. It can be used to Embed Tweets, Timelines, and more within your website/application. Twitter API platform offers three tiers of search APIs: Standard, Premium, and Enterprise. By analyzing social media posts, product reviews, customer feedback, and NPS responses (among other unstructured data), businesses can understand how their customers feel about their product or service.[10]

Sentiment analysis is particularly useful for social media monitoring because it goes beyond the number of likes or retweets, by providing qualitative insights.

Twitter API, comprises of a variety of tasks such as searching tweets, collecting tweets, getting timelines for the tweets, etc. Search APIs offered are 'Standard' which offers retrieval of tweets published in the last 7 days. The 'Premium' tier offers a collection of data from the last 30 days and provides access to the data for better fidelity. and last but not least The 'Enterprise' tier of search API offers data extraction from the last 30 days to data since 2006.

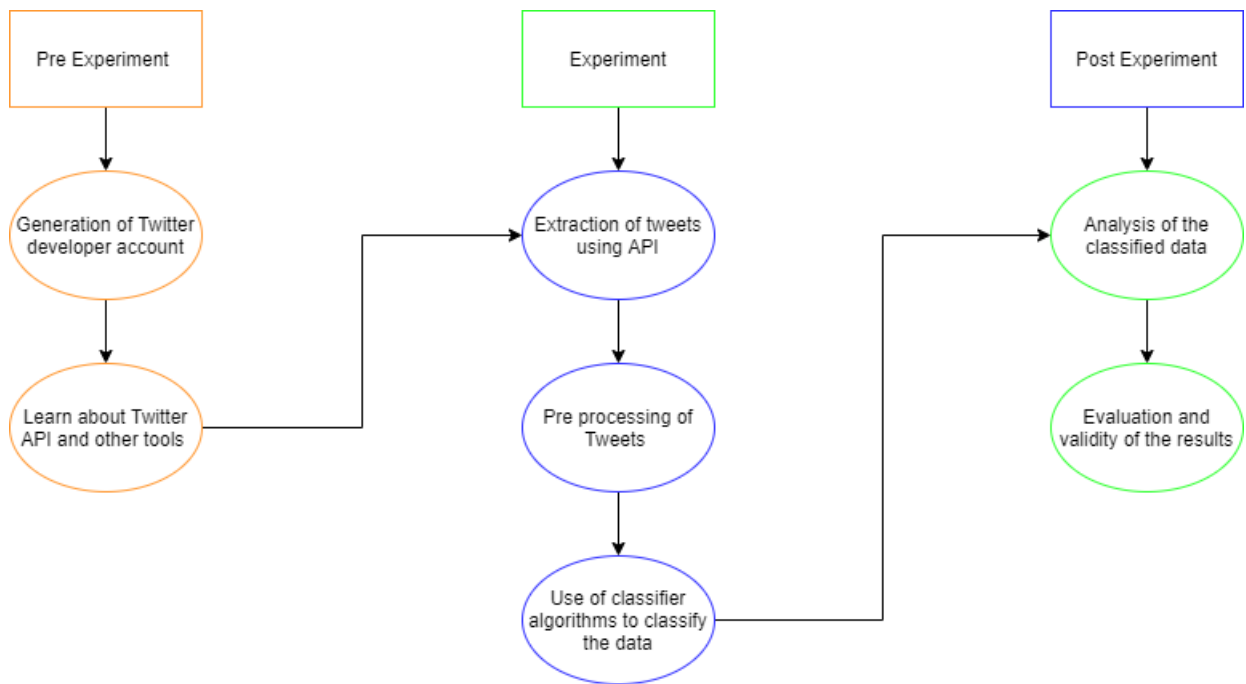


Figure 3.1: Experiment Activities

3.5 Proposed Design

The design of the experiment includes numerous steps from the creation of the Twitter developer account until the evaluation of the results.

The fig. 3.1 shows the activities performed during the study. The steps are as follows:

- **Generation of Twitter developer account:** This step includes the creation of a Twitter developer account for the experiment. Twitter’s Developer Platform enables you to harness the power of Twitter’s open, global, real-time, and historical communication network within your applications. The platform provides tools, resources, data, and API products for you to integrate, and expand Twitter’s impact through research, solutions, and more. This section can help you get acquainted with the current platform organization, explains how to get access to the different tools and endpoints, and provides additional resources that can help you build with the Twitter API.[11]

- **Learn about Twitter API and other tools:** The study of the API and its applications concerning data analysis is to be studied in the next step. The main objective of this step is to understand the extensive applications of Twitter API and how it is related to the experiment performed.
- **Extraction of tweets using API:** The next step is to extract data with the help of Twitter API. The data is obtained by using various keywords and hashtags. The data obtained is in the form of Tweets, Retweets, and comments posted by users on specific topics searched by keywords.
- **Pre-processing of Tweets:** Then the data is passed through the preprocessor to remove unwanted hashtags, slang words, emoticons, and hyperlinks. This filters the data required for the experiment.
- **Use of classifier algorithms to classify the data:** After pre-processing of the data, the sentiment is detected using feature vectors like a bag of words. The use of features helps to identify the sentiment of each sentence as the number of characters allowed on Twitter posts is only 280.
- **Analysis of the classified data:** After sentiment detection, the text is classified into positive, negative, and neutral with the use of classifiers like Support Vector Machine, Naïve Bayes.

- **Evaluation and validity of the results:** The classified data is then analysed on the parameters of accuracy, cross-validation accuracy achieved by each classifier, validity of the model. The results obtained are presented as a goal achieved by the experiment.

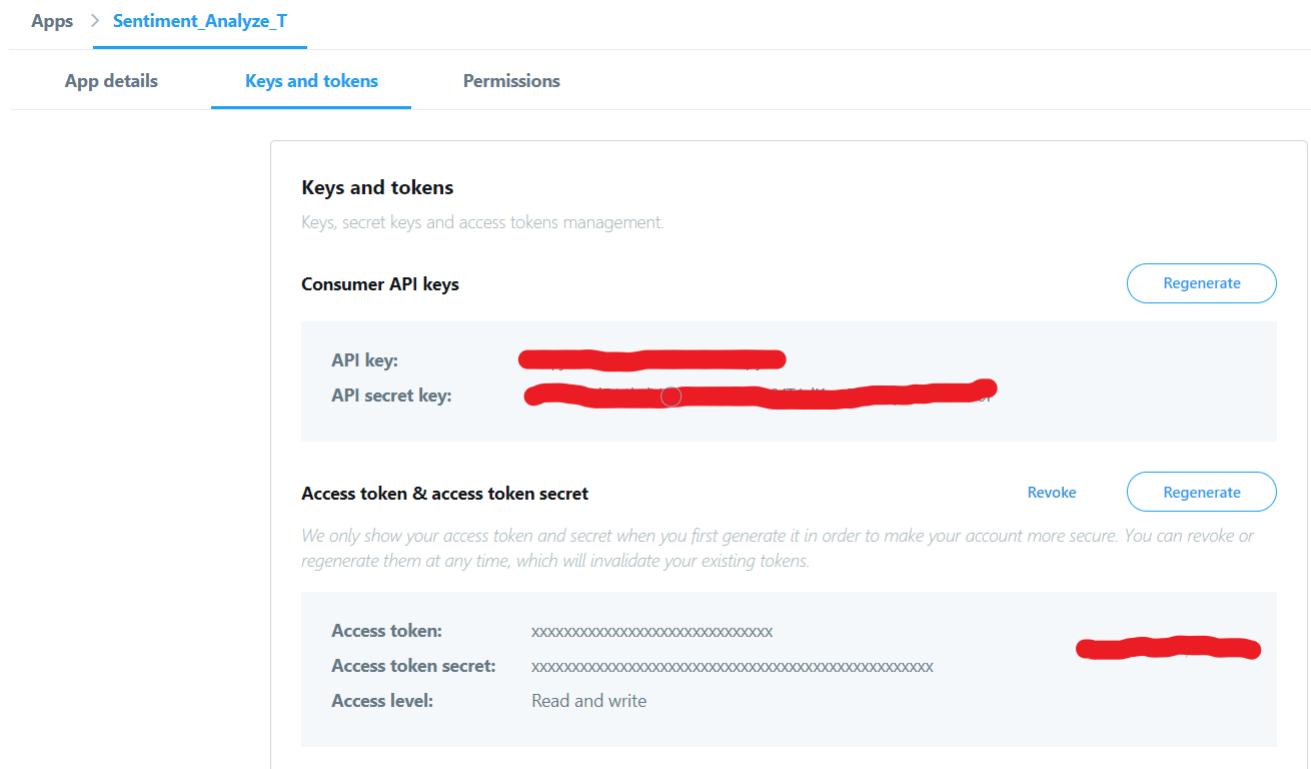


Figure 3.2: API interface with authorization

3.6 Twitter Authorization

To use Twitter API, the user has to register with the developer account. After the generation of the developer's account, the API provides 4 elements for user authorization: API secret key, API key, Access Token, and Access token secret. Without the 4 attributes, the user is not able to make full use of Twitter's API service.

The application interface with all the attributes for this project can be seen in Fig. 3.2.

Metrics for quantitatively assessing the efficiency of message recovery, which is an important aspect to assess the system capability of recovering all the available tweets, due to the several limitations imposed by the Twitter APIs; Minimization of searches on Twitter: query optimization is a critical part of the functional flow of a Twitter analytics platform. Query the parser must take account of Twitter API limit and issue an evaluation plan to minimize twitter API requests[4].

3.7 Data Collection

We are fetching data from twitter using Twitter API. This data is present in raw form that needs to be processed to get meaningful data out of it[3]. The data generated from API contains all types of information which includes hashtags, spaces, punctuation marks and other unnecessary words.

3.8 Data Preprocessing

As the text generated by API is highly unstructured, it needs to be filtered and cleaned before the data analysis. This involves the removal of all unwanted things like removal of URL, hashtags, target, repeated tweets, stop words, blank spaces, and so on.

In phase I tweets are collected by taking the input in the form of hashtags and the number of tweets to be considered is restricted between 5 and 1000[7]. In preprocessing, each word is separated i.e. string sequence is torn into small fragments such as phrases, symbols, and other elements that play a vital role in removing unnecessary information from the tweet. To obtain an optimum decision for the experiment, stop-words are removed. After the removal of stop-words, all the expressions, emoticons are shortened.

After this procedure, the resultant text is then subjected to classifiers and make it identifiable to the classifier[8].

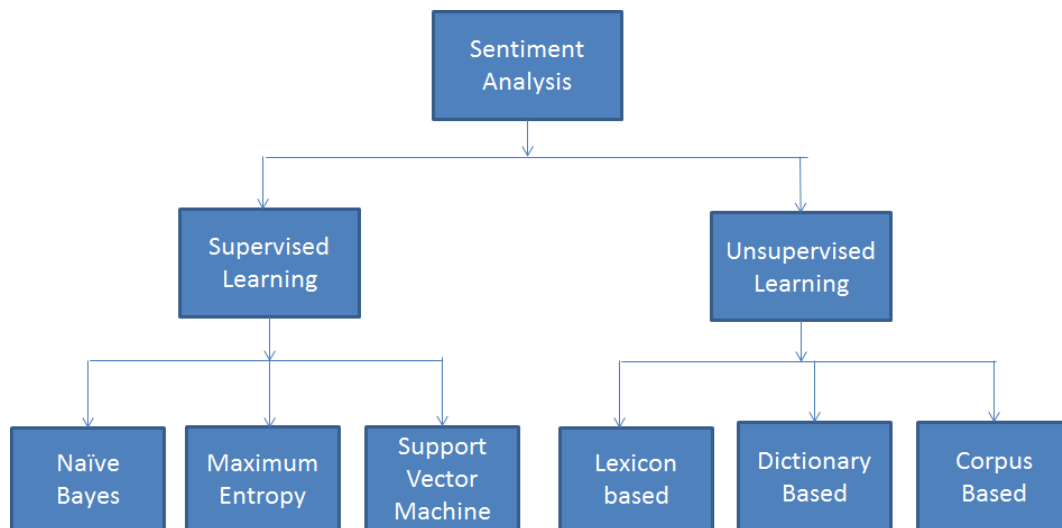


Figure 3.3: Sentiment Classification Algorithms[5]

3.9 Classifiers

After preprocessing of the data is done, the next step is to use classifiers. In this project, we require all the tweets in English. So we have used a Google Translator API to translate all the tweets extracted into English. One of the classifiers i.e. Naive-Bayes Classifier is used to train a small amount of data for the collection of the parameter required for prediction purposes.

Figure 3.3 shows the classification algorithms used for sentiment analysis. It is divided into two types to train the classifiers which consist of Supervised Learning and Unsupervised Learning approach. Supervised learning is further divided into Naive - Bayes, Maximum Entropy, and Support Vector Machine classification algorithm.

Unsupervised learning is further divided into Lexicon based, Dictionary-based and Corpus-based algorithm. We have used supervised learning approach for the experiment.

The diagram shows the Bayes Theorem equation: $P(c|x) = \frac{P(x|c)P(c)}{P(x)}$. Arrows point from the terms to their respective labels: 'Likelihood' points to $P(x|c)$, 'Class Prior Probability' points to $P(c)$, 'Posterior Probability' points to $P(c|x)$, and 'Predictor Prior Probability' points to $P(x)$.

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

Figure 3.4: Bayes Theorem

3.9.1 Naive - Bayes Classifier

It is a probabilistic classifier which uses the properties of Bayes theorem assuming the strong independence between the features. The main benefit of this classifier is that it uses a small amount of training data for the calculation of parameters for the prediction phase. It assumes that the predictors are independent of each other. We classify tweets using the same features as we used the training dataset. Class probabilities for the polarities are calculated using the Naive Bayes logarithmic probabilities. The classifier then assigns the class label to the given tweets.

In simple words, according to the Naïve Bayes classifier, the presence of a particular feature of a class is not related to the presence of any other feature. Say for a given document 'x' and a class 'c' (positive, negative), the conditional probability for each class is $P(c/d)$. The Bayes theorem is shown in Fig. 3.4

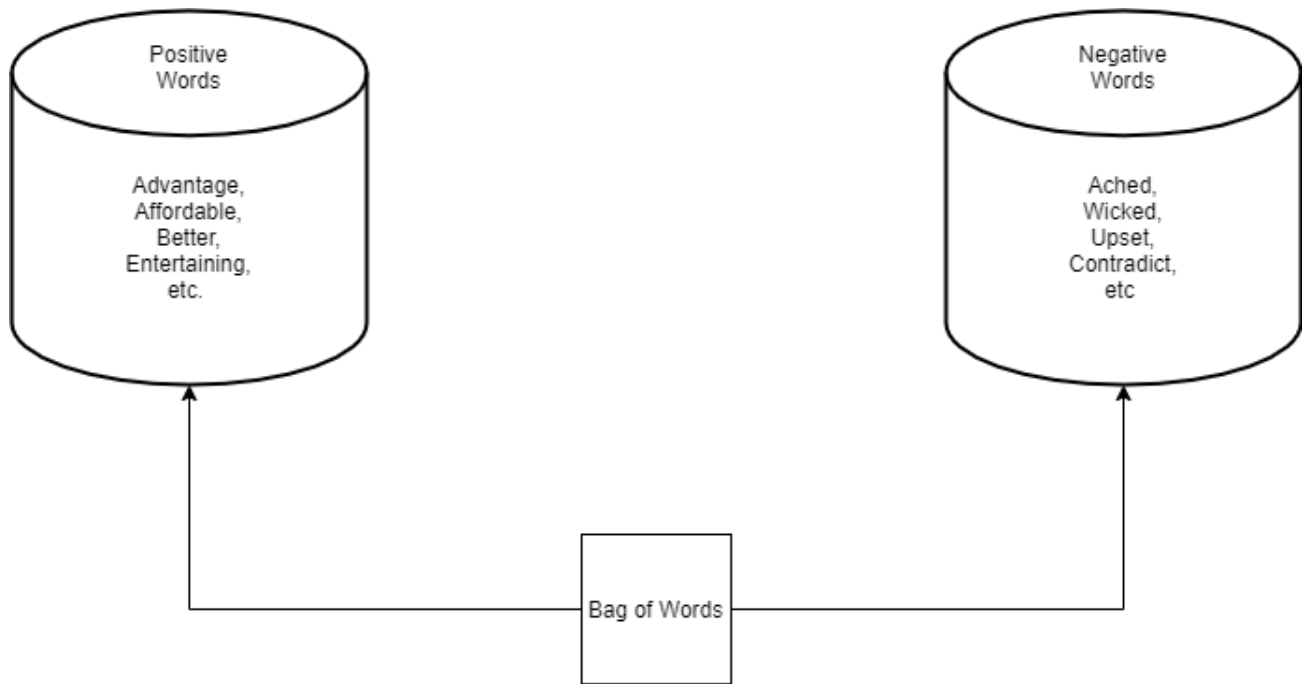


Figure 3.5: Bag of Words

3.9.2 Bag of Words Classifier

This algorithm calculates the frequency of the word observed in the text(tweet). Those word counts let us to evaluate documents and determine their similarities for applications like search, classification of documents and topic modeling[7]. Bag of words contain all the words with positive and negative tone. Few examples of the words are shown in the fig. 3.5.

3.10 Conclusion

A blueprint experimental design was developed in a goal-oriented fashion. Evaluation of validity found a low level of external validity, which is acceptable considering the intended contribution to providing a blueprint for future experiments.

The validity seems to be strong as long as classification algorithms produced desired results. Despite good efforts, the accuracy obtained is considered to be mediocre.

Chapter 4

Execution

4.1 Introduction

This chapter describes each step in the production of the results in the study. It describes steps taken to execute the experiment including the data collection methods, programming tool used, classification algorithm used, and use of internal libraries available in the programming language.

4.2 Software Installation

We have designed our application to perform sentiment analysis. We have used Python as a programming tool for the analysis. Python has various libraries which provide user to perform data analytics of Twitter data. The first step is to install python into the system. After python installation, an IDE is stored in the system. For the experiment, Pycharm IDE is used.

4.3 Tokenization

In an implementation, the live twitter data is classified based on the sentiments. The data is collected by taking the input in the form of hashtags and number of tweets to be considered is restricted between 5 and 1000. The tweets collected are the ones which are streaming live online. In the next step, the collected tweets are pre-processed. Each word in the tweet is tokenized.

Tokenization refers to the act of breaking up a string sequence into pieces such as phrases, words, keywords, symbols and other elements called tokens which plays an important role in removing the unwanted words in the text, like removing special symbol associated with username and hashtag in tweets. Stop words being words which don't alter the meaning of the the sentence is removed, which also minimizes the the effort of classifying every word of tweet by reducing the number of words to be compared. Stemming is performed to reduce words to unify across the documents and make it easier to classify similar words under a category. Cleaning of tweets is done by removing additions of text in the tweets like URLs, numbers, and special characters which shortens the size of tweets for comparison. In phase III, the pre-processed tweets are compared with the available bag of words (BoW) and are classified as positive, negative, and neutral. Two files under a bag of words are used, one for positive and one for negative. An algorithm called Bag of Words which calculates how many times a word appears in a document is used. Those word counts let us evaluate documents and determine their similarities for applications like search, classification of documents, and topic modeling. If the words in the tweet match the words in the positive Bag of words then it is classified as positive tweet, similarly if the word in the tweet matches the words in the negative Bag of words, then it is classified as a negative tweet. And by matching the different categories as mentioned, it is identified to which "bag" a certain block of the text belongs to[7].

4.4 Preparation

Tweepy is a python library that allows users to connect to Twitter API. It supports Python version 2.6, 3.3, 3.6. To install tweepy, I have used 'pip install tweepy' into the command prompt to install it in the system.

After installation of the 'Tweepy' library, it needs to be imported into the python program using `import tweepy` command.

```
import tweepy

access_token = [REDACTED]
access_token_secret = [REDACTED]
consumer_key = [REDACTED]
consumer_secret = [REDACTED]

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

api = tweepy.API(auth)
```

Figure 4.1: Tweepy Implementation

4.4.1 Tweepy

Tweepy provides a connection between the API and python IDE. The code snippet can be seen in Fig. 4.1

This example will provide and print each one of their texts to the console. Twitter requires all requests to use OAuth for authentication. The Authentication Tutorial goes into more details about authentication.

4.4.2 Textblob

Textblob is a python library for processing the textual data generated from Twitter API. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

The same procedure is used to install the textblob library as for tweepy.

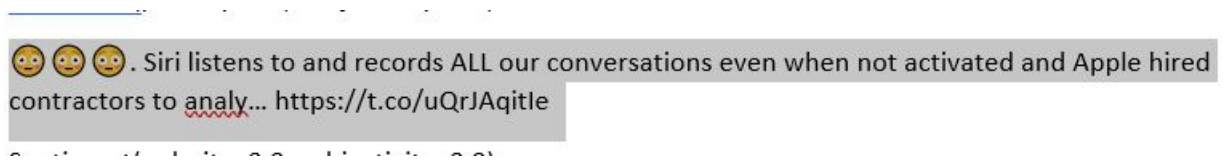


Figure 4.2: Tweet Example

4.5 Data Preprocessing

After the execution of the program, few results are obtained. After typing the keyword into the search command, we have retrieved multiple tweets which contains unwanted data which needs to be cleaned using preprocessing. Example of tweet is seen in Fig. 4.2.

As you can see in the tweet, it contains emoticons, stop-words, and other unwanted elements that need to be filtered for the analysis purposes. We extracted text from tweets and convert it to data frame, removed URLs from text, removed stop words like (the, a, to...), usernames and accounts, removed numbers and unnecessary spaces, and converting encoding (Emojis) from latin1 to ASCII.

After this step, all the punctuation is removed, except for apexes, because they are part of grammar constructs such as the genitive.

The next operation is to remove the vowels repeated in sequence at least three times, because by doing so the words are normalized: for example, two words written in a different way (i.e. coooooool and cool) will become equals. Another substitution is executed on the laughs, which are normally sequences of “a” and “h”. These are replaced with a “laugh” tag.

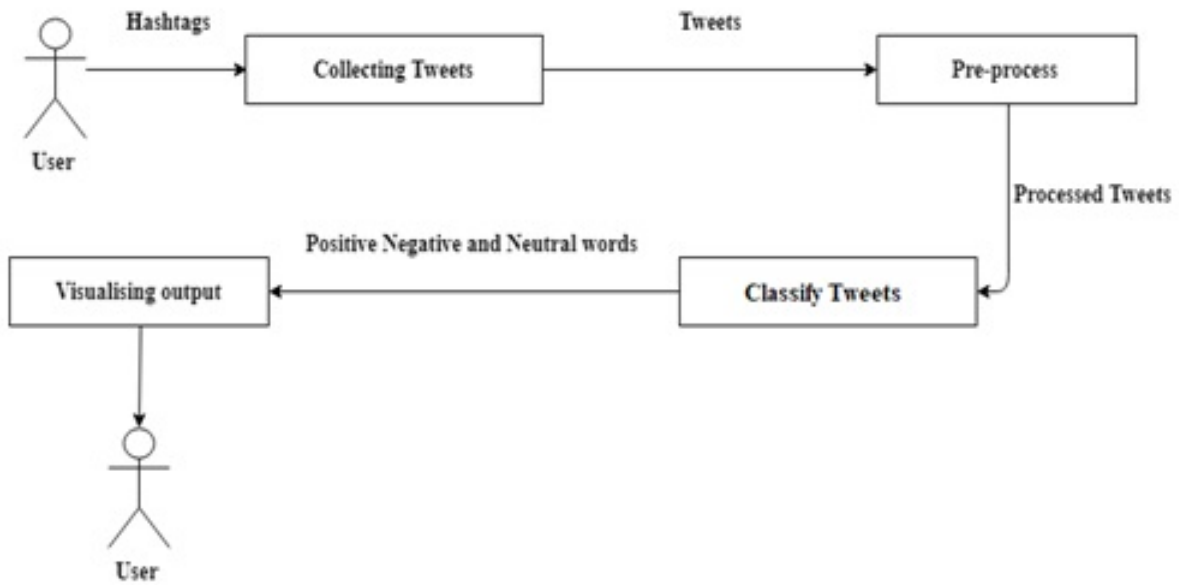


Figure 4.3: Framework of sentiment analysis[7]

All the operations in this module are executed to try to make the text uniform. This is important because during the classification process, features are chosen only when they exceed a certain frequency in the data set. Therefore, after the basic preprocessing operations, having different words written in the same way helps the classification.

smile positive	smile negative
0:-)	ǐ:(
:)	;(
:D	ǐ:)
:*	D;ǐ
:o	:(
:P	:—
;)	ǐ:/

Table 4.1: List of substituted emoticons

4.6 Emoticons

This module reduces the number of emoticons to only two categories: smile positive and smile negative, as shown in Table 4.1.

With the use of feature extraction, the likelihood of some emoticons is found to be used before the classification phase.

4.7 Framework

As shown in the sentiment analysis framework in Fig 4.3, the next step after pre-processing the data, the tweets are then passed through different classifiers for classification into positive, negative, and neutral tweets. The example of the Tweet before cleaning and after cleaning is observed in Fig, 4.4. The emoticons are removed along with the username and URLs to ease the classification phase.

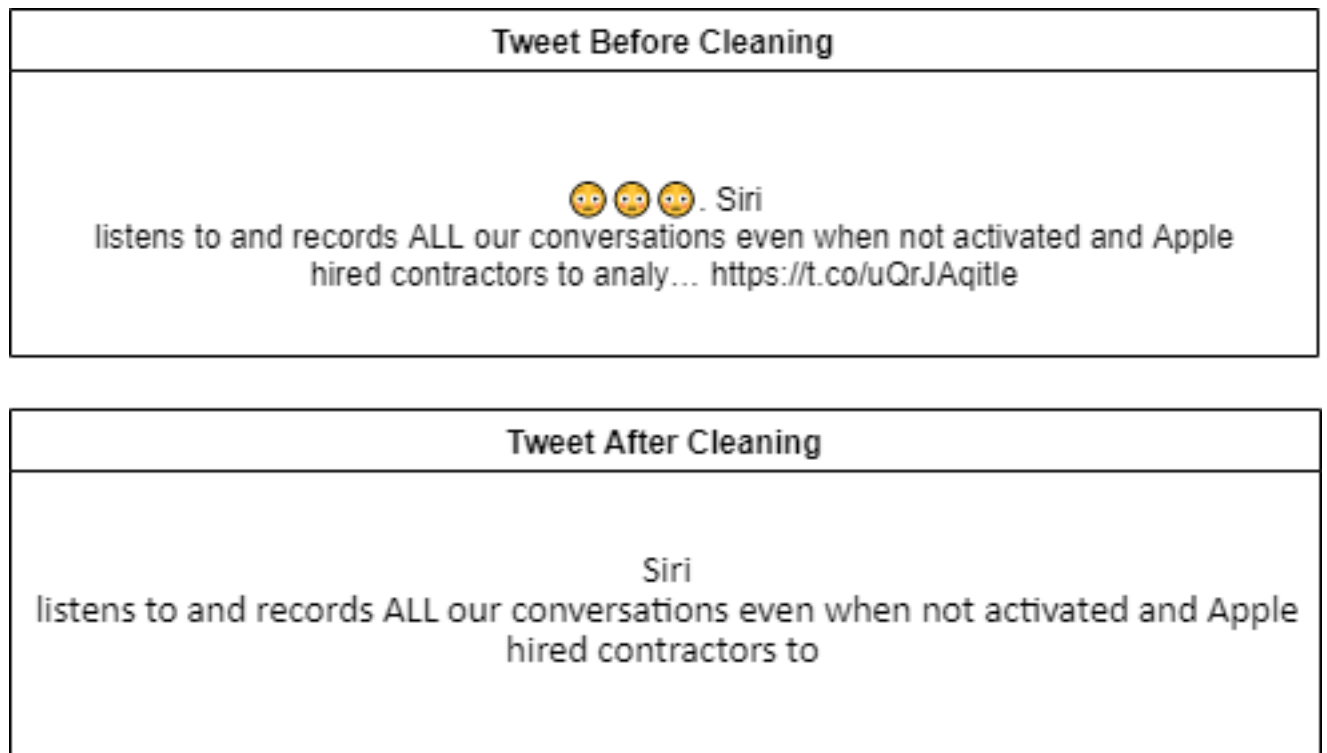


Figure 4.4: Tweet about Apple

4.8 Model Execution

Filtering of the data is multi-step process which includes collection of data and processing it to get the filtered useful information that is much needed by the organizations for the analysis purpose[3]. The tweets generated are classified according to their polarity. If polarity is -1, then it considered to be 'negative'. If the polarity is +1, then sentiment detected to be 'positive' and if it is 0, then it is a 'neutral' tweet. In this phase, after preparing tweet (removing unnecessary symbols), each tweet was labelled as 1, -1, 0. (That's it: positive, negative, or neutral) using unsupervised learning algorithm. Since we do not have pre-classified data, a lexicon-based model can be used to classify tweets. By using two text files containing a list of positive and negative words, along with more words related to our domain. Each word within each tweet is compared to positive and negative documents in order to find matching words, and classify tweets whether it has more positive or negative words.[6]

Long before the data collection, preliminary experimentations were done using classifiers like Artificial Neural Network, Naïve Bayes Classifier[8], and a bag of words classifier. As mentioned in chapter 3, supervised and unsupervised approach is used for sentiment analysis. Using an unsupervised approach, a bag of words classifier is applied to the model.

The dataset used for the bag of words classifier contains a list of positive and negative words which helps to differentiate the tweets according to the tone of the sentence and comparing each word/string to the dataset of the words. Some of the words in the dataset are mentioned below.

List of positive words:

breeze

bright

brighten

brighter

brightest

brilliance

brilliances

brilliant

brilliantly

brisk

brotherly

bullish

Positive Tweet	Feature Words
Breaking bad was always special to me	Special

Figure 4.5: Example showing tweet and feature words

List of negative words:

negate

negation

negative

negatives

negativity

neglect

neglected

negligence

negligent

nemesis

nepotism

This is the tweet posted by official Netflix account : What happened to Jesse Pinkman?
El Camino: A Breaking Bad Movie.

The tweet contains very less number of feature words. So it is difficult to find the tone of the sentence whether its positive or negative.

As compared to the previous example, another tweet posted by another user is shown in Fig. 4.5. The tweet contains the feature word, 'special' which is listed under the positive words dataset. So, the sentiment analysis of the tweet is established as 'positive'.

4.9 Use of Textblob

Decision Tree Classifier is one of the prominent classifiers used in sentiment analysis. Decision Tree classifier in Textblob [11] library is used to classify the text extracted. Using this classifier, the polarity and subjectivity of sentiments from each tweet are calculated. From the scores of polarity and subjectivity, the tweet is classified as positive, negative, or neutral[8]. The features of the textblob are noun feature extraction, sentiment analysis.

The command used for collection of tweets using search command,

```
public_tweets = api.search('Breaking Bad Netflix').
```

This command allows the program to find public tweets posted by users using keywords entered in the api.search. The results are obtained in the form of tweets, subjectivity, and polarity.

```

Top best TV shows on Netflix that you can see with your Parents.
. FRIENDS
. LUCIFER
. BARD OF BLOOD (violence)

```

Figure 4.6: Sample Tweet about 'Bard of Blood'

4.9.1 Polarity

The output is generated generating sentiment in terms of polarity which varies in along three factors -1, 0 and +1. The example is as follows.

What happened to Jesse Pinkman?

El Camino: A Breaking Bad Movie

October 11

```
Sentiment(polarity=-0.6999999999999998,
subjectivity=0.6666666666666666)
```

Polarity detection is done by using different lexicons. Here in the retrieved tweet, the polarity is observed to be -0.699999 which is near to -1. Using the logic, the tweet is termed as 'negative' after sentiment analysis of the tweet.

4.9.2 Subjectivity

The results are obtained in terms of subjectivity along with polarity. Subjective sentences generally refer to personal opinion, emotion, or judgment whereas objective refers to factual information. Subjectivity is also a float which lies in the range of [0,1].

Another TV show is searched through the api and output obtained after the use of sentiment function is seen in Fig 4.6 and 4.7.

```
Sentiment(polarity=0.75, subjectivity=0.4)
```

Figure 4.7: Sentiment Function Results

The subjectivity of the tweet is 0.4 as seen in Fig. 4.7. Subjectivity refers to the personal opinion of the user rather than focussing on the factual information. It helps in extracting sentiment of the user about the TV show 'Bard of Blood'.

```
from google.cloud import translate_v2 as translate

translate_client = translate.Client()

if isinstance(text, six.binary_type):
    text = text.decode("utf-8")

# Text can also be a sequence of strings, in which case this method
# will return a sequence of results for each text.
result = translate_client.translate(text, target_language=target)

print(u"Text: {}".format(result["input"]))
print(u"Translation: {}".format(result["translatedText"]))
print(u"Detected source language: {}".format(result["detectedSourceLanguage"]))
```

Figure 4.8: Text translation JSON

4.10 Google Translator API

The tweets generated are in multiple languages. Those tweets are classified and then translated to English using Google translator API for python. So the results obtained are solely in English. The JSON used for text translation is shown in the fig 4.8.

The text obtained is converted into English language using translate API for Python and then subjected to sentiment function to detect the tone of the tweet.

Chapter 5

Analysis

5.1 Introduction

This chapter summarizes the data collected and the treatment of data. The results obtained from sentiment analysis are analyzed based on the validity of the model and accuracy achieved from the classifiers.

5.2 Technological Products

The data collected via Twitter API after performing all the operations as mentioned in Chapter 4 needs to be analyzed. People posted few tweets about Cisco products. The number of tweets generated for the analysis purpose are 540.

Number of Tweets collected	Product
540	Cisco

Table 5.1: Cisco



Figure 5.1: Visualization of Tweets about Cisco

After performing sentiment analysis using classifiers, the results obtained are seen in Fig. 5.1. In the figure, the number of people who feel positive about the Cisco products is more than negative sentiment. The two classifiers used to identify the sentiment of the tweets were Naive Bayes and Decision Tree.

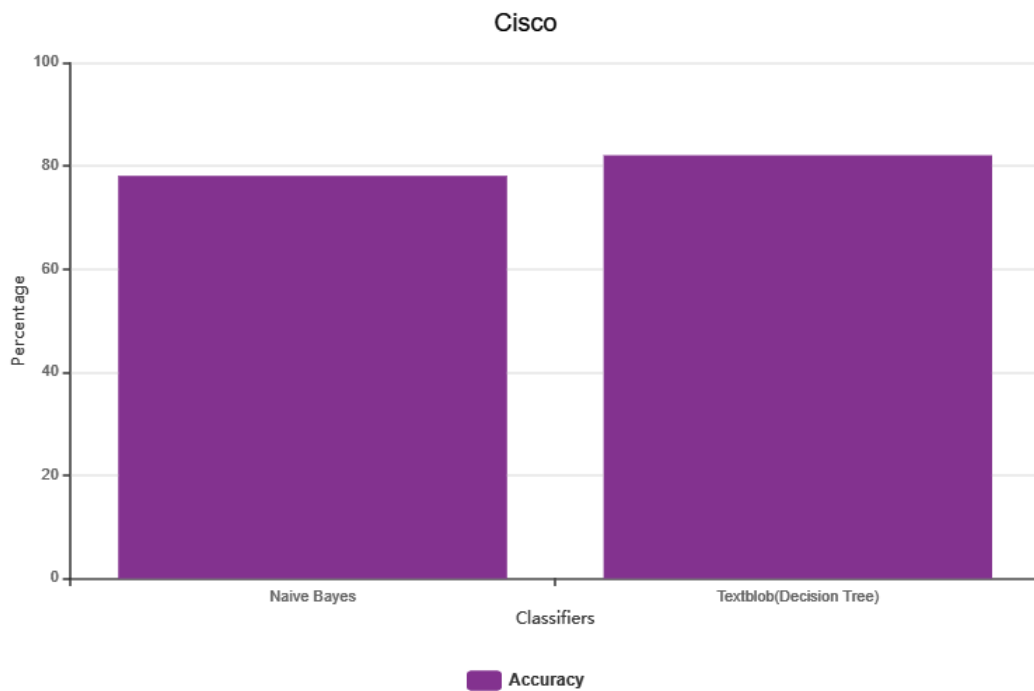


Figure 5.2: Visualization of Tweets about Cisco based on precision matrix

For data visualization, a Python tool matplotlib to compare two classifiers based on the accuracy of the model. The results obtained are shown in Fig. 5.2.

As observed in the bar graph generated, the accuracy achieved by the Naive Bayes classifier is 78%, and the accuracy achieved by the Decision tree classifier which is used in the Textblob library of python is 82%. The Naive Bayes classifier uses probability of feature words in tweet with respect to overall training dataset feature words and the precision is obtained to be 78%. But in another case of decision tree classifier, Tagging text by sentiment is highly subjective, influenced by personal experiences, thoughts, and beliefs. Subjectivity is another major parameter in Textblob classifier which increases the accuracy of the model. According to the analysis, higher accuracy is achieved by the Decision tree classifier to classify the data based on the sentiment of the tweet posted by the user. The accuracy is determined by use of features.

The fundamental thought of sentiment analysis is to change unstructured data into significant or meaningful data. After performing sentiment analysis on Tweets about Cisco

products, the classifier used for Huawei phones, one of the largest manufacturer of Smart-phones in the world.

Number of Tweets	Product(Huawei)
400	Huawei P30 pro
230	Huawei Talkband 3
310	Huawei Matepad T 8

Table 5.2: Huawei Products

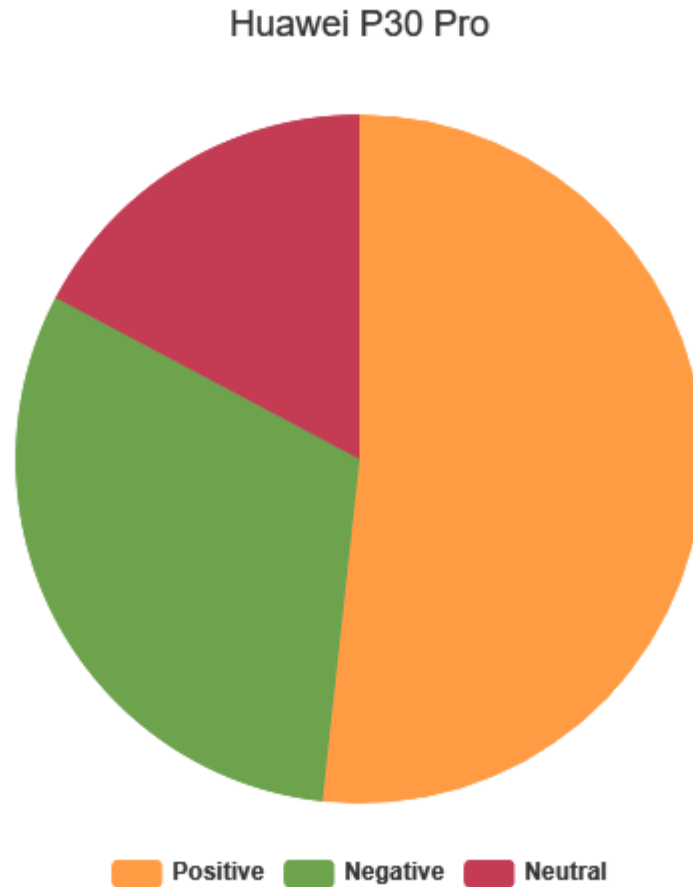


Figure 5.3: Pie Chart representing Sentiment Analysis of Huawei P30 pro

Sentiment Analysis is performed on the smartphones and other products launched by Huawei.

Table 5.2 contains information about the number of tweets generated along with the name of the product. For Smartphone P30 pro, users have posted opinions on Twitter after using the phone for a suitable amount of time. Number of tweets used for the study purposes are 400. Out of 400 tweets, the positive sentiment is observed in 230 tweets which is 57.5 percent.

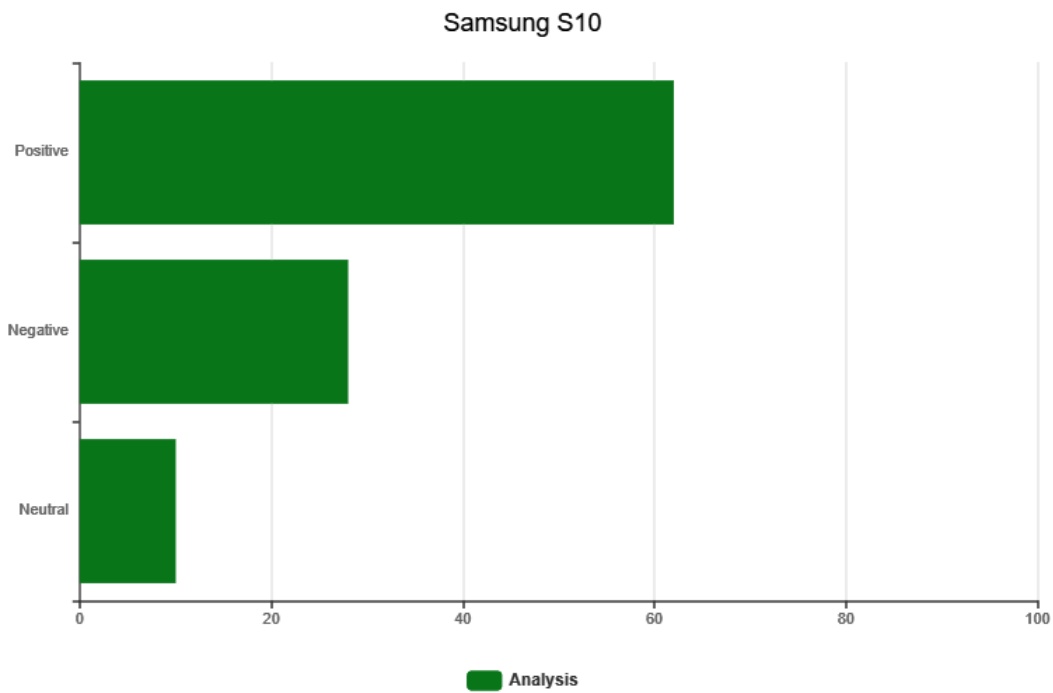


Figure 5.4: Horizontal Bar graph for Sentiment Analysis of Samsung S10

Another analysis was done on the Samsung S10 smartphone device. User experience is posted via tweets. The expressed opinions about the product are analyzed and visualized with a bar graph as seen in Fig. 5.4. As we obtain the information from the bar graph, amongst the tweets, 62% of users liked the smartphone. So the sentiment is termed as 'positive'. 28% people were not happy with the product while 10% of people are not having either a positive or negative review about the product. So, it is stated as 'neutral' sentiment.

5.3 Analysis of Web Series and TV series

Numerous movies and TV series release every year in the world having various genre like action, comedy, romantic. People express their views on the released movies via Twitter. Data is collected for such opinions for opinion mining or sentiment analysis purpose.

WHO (World health organization) declared COVID-19 as a pandemic in mid-march of 2020. The whole world went into lockdown and was advised to stay home for the safety of the people. People started consuming a lot of content while staying home. The number of users has increased since then on Twitter and people started posting their opinions or views on Twitter about the content consumed on daily basis.

The OTT(Over the top media service) platforms like Netflix, Amazon Prime Video started releasing various Web series and movies on the platform. After consuming the content, people posted their views on Twitter whether they like the show/movie or dislike it. The tweets are collected for such web series, TV series, and Movies for sentiment analysis. A few of the examples and analysis are mentioned in this section.

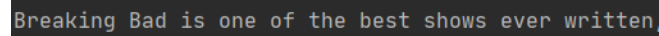


Figure 5.5: Tweet about Breaking Bad

Tweets Collected	TV Show
438	Breaking Bad

Table 5.3: Breaking Bad

The sample tweet posted by user is shown in Fig. 5.5. The text in the image implies that the user liked the show 'Breaking Bad' and is considered to be positive sentiment for the show. The sentiment analysis for the tweet is emerged as 'Positive'.

The number of Tweets collected for opinion mining or sentiment analysis is mentioned in table 5.3. Out of which most of the tweets are generating positive sentiment about the TV show. After sentiment analysis of each tweet collected, 88% users liked the show, and 10% users didn't like the show, and 2% users posted only about watching the show and not sharing personal opinions about the show. Data visualization is performed and seen in Fig. 5.6.

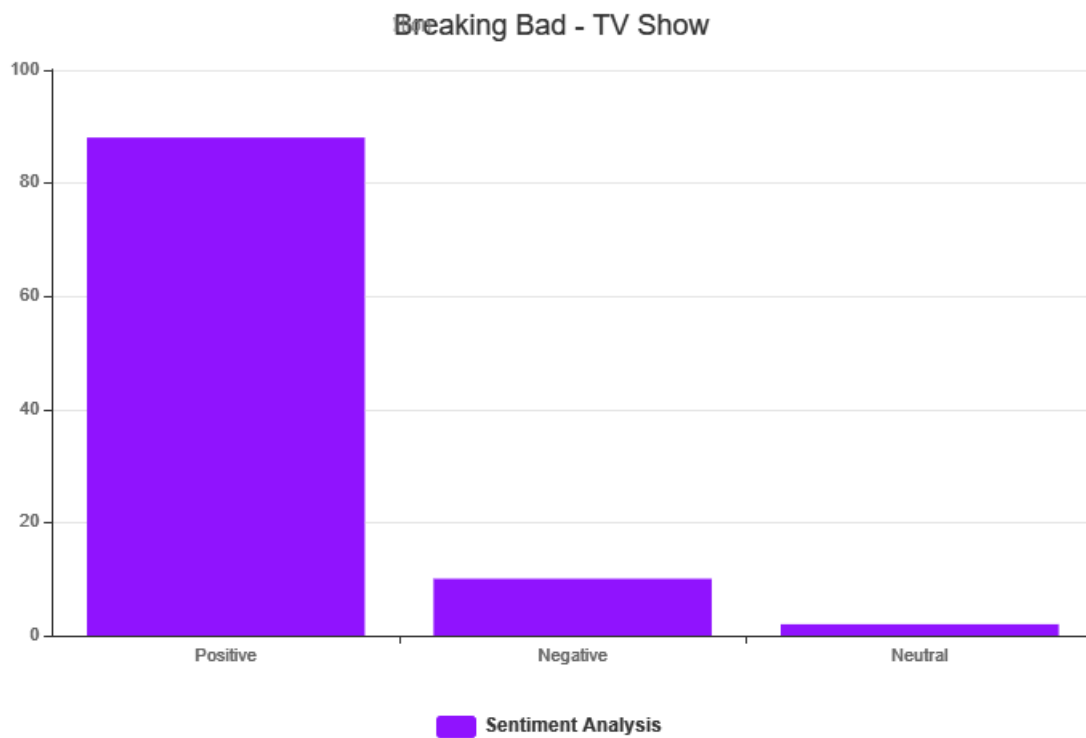


Figure 5.6: Breaking Bad

One of the major OTT platforms Netflix released the most awaited season of 'La Casa De Papel' or 'Money Heist' during the lockdown period. As people were confined in their respective homes, they have watched all the previous seasons and new season during the confinement and posted their views on Twitter. Number of Tweets collected from the Twitter API are 254. A brief comparison of previous seasons and new season is performed during analysis and can be visualize with the help of bar graph generated in Fig. 5.7.

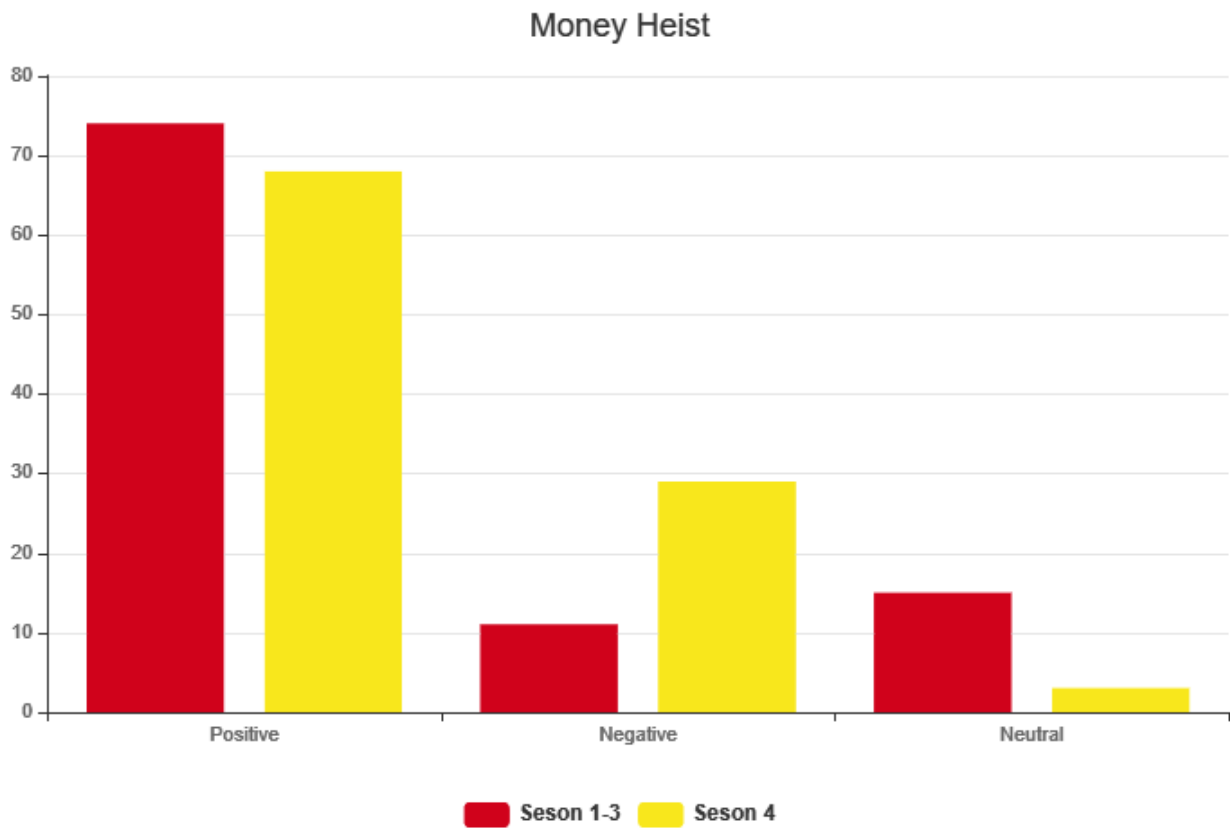


Figure 5.7: Money Heist

The positive reviews have dropped from 74% to 68% from season 1-3 to season 4. Sentiment analysis shows that negative sentiment is increased for the new season as compared to previous seasons. The percentage of negative sentiment is escalated from 11% to 29%.

Day	Tweets Collected	Web Series
1	20	Upload
2	58	Upload
3	47	Upload
4	88	Upload
5	71	Upload
6	108	Upload

Table 5.4: UPLOAD Web Series

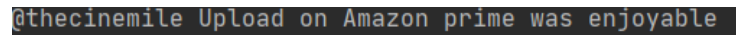
A screenshot of a tweet from the user @thecinemile. The text of the tweet is "Upload on Amazon prime was enjoyable". The text is displayed in white on a dark background, which is typical for a tweet screenshot.

Figure 5.8: UPLOAD

Amazon Prime Video has released its new Amazon originals web series called 'Upload'. After consuming the whole season, users have posted opinions about the show. The statistics of the data collection is in Table 5.4. An example of tweet is seen in Fig. 5.8.

After collecting the data i.e. tweets, the actions were performed on the data for analysis. The sentiment analysis of the show for each day turned out to be different. Users have posted vivid tweets after watching the show. The analyzed data then observed through the bar graph. The bar graph is shown in Fig. 5.9 represents sentiment analysis observed on each day.

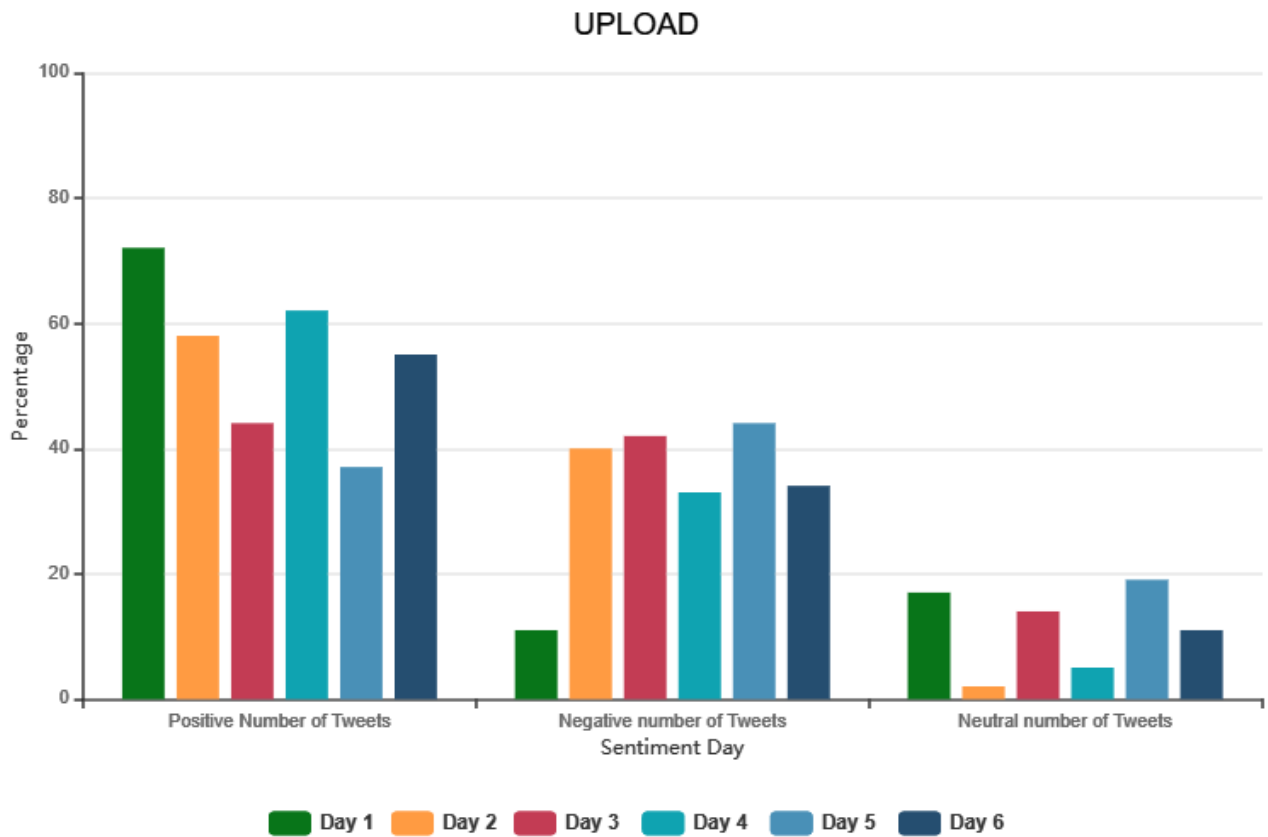


Figure 5.9: UPLOAD review for each day

5.4 Movies

People have re-watched old movies as no new movies were released. All the theatres were shut down due to COVID-19. New user opinions are generated from Twitter. These opinions were used to identify sentiment for each tweet.

Sentiment analysis is performed on few of the movies. The list of movies used for database are stated in the table 5.5.

Release Year	Tweets Collected	Movies
2001	15	Lagaan
2001	29	A Beautiful Mind
2019	118	Avengers: Endgame
2012	19	Balgandharva
2019	154	Joker

Table 5.5: Movies Database

```
Lagaan was a horribly made movie. First half was tolerable. Second half sucked
```

Figure 5.10: Sample Tweet About Lagaan

After watching movies, users have posted views via Twitter. The sentiment analysis is performed on listed movies. A sample tweet posted about the movie Lagaan is stated in Fig. 5.10. The tone of the sentence is negative.

Sentiment Analysis of the data is converted to the bar graph for a better understanding. The bar graph is about 2 Indian movies with a different language.

'Lagaan' received the most negative reviews as observed in Fig. 5.11 compared to 'Balgandharva' movie. The percentage of positive reviews generated is 42 and 88 for both the movies respectively.

Avengers: Endgame became the most profitable movie released all time. But the movie received some negative reviews on Twitter by few of the users. The number generated after analysis is stated in table 5.6.



Figure 5.11: Sentiment Analysis of Indian Movies

Sentiment	Number of Tweets	
Positive	100	Avengers: Endgame
Negative	17	Avengers: Endgame
Neutral	1	Avengers: Endgame

Table 5.6: Reviews for Avengers: Endgame

5.5 Conclusion

The results are obtained using the Decision Tree classifier and as previously mentioned, the accuracy is observed to be 89%. The final results after sentiment analysis of technological products, Web/TV series, and movies are visualized in the form of bar graphs and pie charts.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This research aimed towards the study of Opinion Mining or Sentiment Analysis of text. Sentiment Analysis is a field of study based on opinions expressed in text in various industries. The social media aspect of sentiment analysis is focused on this research.

In this study, Twitter is used for text classification in the format of tweets posted by users demonstrating opinions about certain products, movies, web series, and TV series. It uniquely displays the results in graph form as well as in text form for detailed extraction of user's opinions. A systematic analysis of the tweets is performed to give results from Twitter. The model was proposed with the use of different classifiers to enhance the results of the classification of tweets as positive, negative, and neutral. Two of the classifiers Naive Bayes and Decision Tree were used in this study.

For testing various metrics were used, and it is shown that based on accuracy, the decision tree classifier has higher accuracy. OTT(Over-the-Top media) platforms were the focus for the collection of movie/web series reviews in the form of tweets posted by users. The results of sentiment analysis were exhibited in pie charts, bar graphs.

6.2 Future Work

Future work can include testing of the model on discrete tweets posted by users in various fields like rumor detection about the specific entity to avoid misjudgment of the people. Some of the more difficult challenges of Sentimental Analysis could also be used for further extensions of this study. The model can be aimed to use emoticons for classification in the future. The results can be synced with the actual frame of mind of user and customer feedback.

Bibliography

1. P. Tripathi, S. K. Vishwakarma and A. Lala, "Sentiment Analysis of English Tweets Using Rapid Miner," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, 2015, pp. 668-672, doi: 10.1109/CICN.2015.137.
2. L. Wang and J. Q. Gan, "Prediction of the 2017 French election based on Twitter data analysis," 2017 9th Computer Science and Electronic Engineering (CEECE), Colchester, 2017, pp. 89-93, doi: 10.1109/CEECE.2017.8101605.
3. A. K. Soni, "Multi-lingual sentiment analysis of Twitter data by using classification algorithms," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, 2017, pp. 1-5, doi: 10.1109/ICECCT.2017.8117884.
4. D. Cenni, P. Nesi, G. Pantaleo and I. Zaza, "Twitter vigilance: A multi-user platform for cross-domain Twitter data analytics, NLP and sentiment analysis," 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, 2017, pp. 1-8, doi: 10.1109/UIC-ATC.2017.8397589.
5. R. Wagh and P. Punde, "Survey on Sentiment Analysis using Twitter Dataset," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, 2018, pp. 208-211, doi: 10.1109/ICECA.2018.8474783.

6. S. A. El Rahman, F. A. AlOtaibi and W. A. AlShehri, "Sentiment Analysis of Twitter Data," 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 2019, pp. 1-4, doi: 10.1109/ICCISci.2019.8716464.
7. V. Prakruthi, D. Sindhu and D. S. Anupama Kumar, "Real Time Sentiment Analysis Of Twitter Posts," 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 2018, pp. 29-34, doi: 10.1109/CSITSS.2018.8768774.
8. F. J. J. Joseph, "Twitter Based Outcome Predictions of 2019 Indian General Elections Using Decision Tree," 2019 4th International Conference on Information Technology (INCIT), Bangkok, Thailand, 2019, pp. 50-53, doi: 10.1109/INCIT.2019.8911975.
9. A. Aslam, U. Qamar, R. A. Khan, P. Saqib, A. Ahmad and A. Qadeer, "Opinion Mining Using Live Twitter Data," 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), New York, NY, USA, 2019, pp. 36-39, doi: 10.1109/CSE/EUC.2019.00016.
10. Twitter API Information, <https://developer.twitter.com/en/docs>
11. Twitter Developer Platform, <https://developer.twitter.com/en/docs>
12. Pinto H.L., Rocio V. (2019) Combining Sentiment Analysis Scores to Improve Accuracy of Polarity Classification in MOOC Posts. In: Moura Oliveira P., Novais P., Reis L. (eds) Progress in Artificial Intelligence. EPIA 2019. Lecture Notes in Computer Science, vol 11804. Springer, Cham.