“TWITTER SENTIMENTAL ANALYSIS”

PROJECT REPORT

Submitted for the course: Cse3019 Data Mining

By

Cheruku Manish 16BCE0538

Vishal Suresh Salvankar 16BCE0723

Slot: F2

Name of faculty: Prof. Ramani S

(SCHOOL OF COMPUTER SCIENCE AND ENGINEERING)

November 2018

CERTIFICATE

This is to certify that the project work entitled **"Twitter Sentimental Analysis"** that is being submitted by **"Cheruku Manish (16BCE0538) Vishal Suresh Salvankar (16BCE0723)"** for CSE3019 Data Mining is a record of bonafide work done under my supervision. The contents of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted for any other CAL course.

Place: Vellore

Date:

Signature of Student:

Signature of Faculty:

ACKNOWLEDGEMENTS

I take immense pleasure in thanking **Dr. G. Viswanathan**, my beloved Chancellor, VIT University and respected Dean, **Dr. R. Saravanan,** for having permitted me to carry out the project.

I express gratitude to my guide, **Prof. Ramani S,** for guidance and suggestions that helped me to complete the project on time. Words are inadequate to express my gratitude to the faculty and staff members who encouraged and supported me during the project. Finally, I would like to thank my ever-loving parents for their blessings and my friends for their timely help and support.

Signature of Student

ABSTRACT

This project deals with the problem of sentiment analysis in twitter; that is classifying tweets according to the sentiment expressed as: positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allowing users to write short status updates with maximum length 140 characters. It's a rapidly expanding service with over 200 million registered users - 100 million of whom are active users and half log on to twitter everyday - generating nearly 250 million tweets per day. Because of this large amount of use, we hope to reflect public feelings by analyzing the feelings expressed in the tweets. Analyzing public sentiment is important for many applications, such as companies trying to find out their products ' response on the market, predicting political elections and predicting socio - economic phenomena such as stock exchanges. The purpose of this project is to extract a Detailed Human Labelling functional classifier of random tweets into categories - Positive, Strong Positive, Weakly Positive, Negative, Strongly Negative, Weakly Negative, and Neutral.

Keywords: Sentimental Analysis, Twitter, Human Labelling, and Classification

TABLE OF CONTENTS

1. INTRODUCTION

1.1 MOTIVATION

We have chosen to work with twitter because we feel it is a better approximation of public sentiment as compared to conventional Internet articles and web blogs. The reason is that the amount of relevant data for twitter is much higher than traditional blogging sites. In addition, the response on twitter is more prompt and also more general (because the number of users tweeting is considerably greater than those who write web blogs on a daily basis).

Public sensitivity analysis is highly critical in macro - economic phenomena such as forecasting a particular company's stock market rate. This could be done by analyzing the overall public feeling towards the company in terms of time and using economic tools to find a correlation between the public feeling and the company's stock market value.

Firms can also estimate how well their product is responding on the market, which areas of the market are responding and where a negative response twitter allows us to download geotagged tweet streams for specific locations.

If companies can obtain this information, they can analyze the reasons behind the geographically differentiated response and therefore market their product more optimizedly by looking for suitable solutions, such as the creation of suitable market segments.

Predicting the outcome of popular political elections and polls is also a emerging application for feeling analysis. Tumasjan et al. carried out one such study. For predicting the outcome of federal elections in Germany, Twitter is a good reflection of offline feeling [1].

1.2 PROJECT DOMAIN DESCRIPTION

This project of the analysis of tweet feelings comes under the domains of " Human Labeling “, “Sentimental Analysis”, and “Data Mining.”

Sentiment Analysis (SA) is the process of calculating whether a text is positive, negative or neutral.

Business: In Marketing Field Companies SA develops its strategies, understands customers ' feelings about products or brands, how people respond to their campaigns or product launches, and why consumers do not buy certain products.

Politics: In the political field, SA is used to track political views, to detect coherence and inconsistency between government statements and actions. It can also be used to predict election results

Public Actions: SA is also used to monitor and analyze social phenomena, identify potentially hazardous situations and determine the overall mood of the blogosphere.

2. LITERATURE SURVEY

2.1 LIMITATIONS

Sentiment analysis of micro blogging is a relatively new research topic in the field, so there is still a lot of room for further research in this area.

A decent amount of related previous work has been done on feeling analysis of user reviews, documents, web blogs / articles and general sentence level feeling analysis [2]. These differ mainly from twitter due to the 140-character limit per tweet, which forces the user to express an opinion compressed in very short text.

The best results reached in sentiment classification use supervised learning techniques such as Naive Bayes and Support Vector Machines, but the manual identification required for the supervised approach is very expensive. Some work has been done non-monitoring ([3] and [4]) and semi-monitoring ([5] and [6]) approaches, and there is a lot of room of improvement.

Different researchers who test new features and classification techniques often compare their results to basic performance. In order to select the best features and most efficient classification techniques for individual applications, proper and formal comparisons between these results are needed.

2.2 RELATED WORK

The bag-of-words model is one of the most widely used feature models for nearly all text classification tasks due to its simplicity and good performance.

The model represents the text to be classified as a bag or collection of individual words without any link or dependence on one word or the other, i.e. it completely ignores the grammar and order of words in the text.

This model is also popular in sentimental analysis. The simplest way to incorporate this model is by using unigrams as features in our classifier. Generally speaking, n-grams is a contiguous sequence of " n " words in our text that is completely independent of any other text words or grams.

Unigrams is therefore only a collection of individual words in the text to be classified, and we assume that the probability of the occurrence of one word will not be affected by the presence or absence of any other word in the text.

This is a very simplified assumption, but rather good performance has been shown ([7] and [8]).

One simple way of using unigrams as features is to assign them to a certain prior polarity and take the average of the total polarity of the text, where the total polarity of the text could be calculated simply by summing the previous polarities of the individual unigrams.

Prior polarity of the word would be positive if the word is generally used as an indication of positivity, for example the word “sweet”; whereas it would be negative if the word is generally associated with

negative connotations, for example “evil”. There may also be polarity levels in the model, which means how much the word for this particular class is indicative. A word such as " awesome " would probably have strong subjective polarity along with positivity, whereas the word " decent " would have positive prior polarity, but probably weak subjectivity.

There are three ways to use word polarity as features. The simpler unmonitored approach is to use publicly available online lexicons / dictionaries that map a word to their previous polarity.

The Multi-Perspective-Question-Answering (MPQA) is an online resource with such a subjectivity lexicon, which maps a total of 4,850 words according to whether they are “positive” or “negative” and whether they have “strong” or “weak” subjectivity [9].

Another resource of SentiWordNet 3.0 is the probability of each word belonging to positive, negative and neutral classes[10].

The second approach is to construct a customized prior polarity dictionary based on the occurrence of each word in each particular class. For example, if a certain word occurs more often in the positive labeled phrases in our training dataset (as compared to other classes), we can calculate that the probability of that word belonging to the positive class is higher than the probability of any other class.

This approach has shown a better performance, since the previous polarity of words is more suited and suitable for a particular type of text and is not very general as in the previous approach. However, the latter is a supervised approach, because training data must be labeled in the appropriate classes before a relative occurrence of a word can be calculated in each class.

Kouloumpis et al. noted a decrease in performance by using the lexicon word features together with the custom n-gram word features built from training data as opposed to when the n-grams were used alone[7].

The third approach is a middle ground between the two approaches above. We build our own polarity lexicon in this approach, but not necessarily from our training data, so we do not need labeled training data.

One way of doing this as proposed by Turney et al. is to calculate the prior semantic orientation (polarity) of a word or phrase by calculating it's mutual information with the word "excellent" and subtracting the result with the mutual information of that word or phrase with the word "poor" [11]. They used the number of result-hit counts from online search engines of a relevant query to compute the mutual information.

The final formula they used is as follows:

$$\boldsymbol{Polarity(phrase)} = \boldsymbol{log_2} \frac{\boldsymbol{hits(phrase\ NEAR\ \text{"excellent"}).hits(\text{"poor"})}}{\boldsymbol{hits(phrase\ NEAR\ \text{"poor"}).hits(\text{"excellent"})}}$$

Where hits(phrase NEAR " excellent ") means the number of documents returned by the search engine in which the phrase (which is to be calculated by polarity) and the word " excellent "are co-curring. While hits("excellent") means a number of retuned documents containing the word " excellent. "

Prabowo et al. used a seed of 120 positive words and 120 negative to perform internet searches[12]. Thus, the overall semantic orientation of the word under consideration can be found by calculating the closeness of the word with each of the seed words and by taking it and the average.

Hatzivassiloglou et al. discussed another graphic method of calculating adjective polarity [13]. The process involves first identifying all conjunctions of corpus adjectives and using a supervised algorithm to distinguish each pair of adjectives from the same or different semantic orientation. A graph is constructed which indicates the same or different semantic orientation of the nodes.

Finally, a clustering algorithm is applied that divides the graph into two subsets so that nodes in a subset contain mainly links of the same orientation and links between the two subsets contain mainly links of different orientation. One subset would contain positive adjectives and the other would contain negative adjectives.

SYSTEM DESIGN

The process of designing a functional classification for feeling analysis may be divided into five basic categories. They're the following:

1. Data Acquisition
2. Human Labelling
3. Feature Extraction

HUMAN LABELLING

This is the method we have used in this report.

We labelled the tweets in 3 classes (rather 7 classes) according to sentiments observed in the tweets: positive, negative, neutral.

We gave the following guidelines to our labellers to help them in the labelling process:

1. **Positive**: If the whole tweet has a positive / happy / excited / joyful attitude or something with positive connotations is mentioned. If more than one feeling is expressed in the tweet, the positive feeling is more dominant. It can further divided into Strongly Positive, Weakly Positive as well. See appendix 2.

2. Negative: If the whole tweet has a negative / sad / displeased attitude or something with negative connotations is mentioned. If more than one feeling is expressed in the tweet, the negative feeling is more dominant. It can further divided into Strongly Negative, Weakly Negative as well. See the appendix 2 .

3. Neutral: If the tweet creator expresses no personal feeling / opinion in the tweet and simply transmits information. See appendix 2.

4. SYSTEM IMPLEMENTATION

Various libraries such as **re, tweepy, textbob and matplotlib** are used in the code.

Re is used to for pattern checking such that a tweet should be in the same pattern.

Tweepy is used to connect Twitter to Python and also for getting the tweets.

Textblob is a Natural Language Processing library, which finds the polarity of the sentence according to the impact of all the positive and negative meaning words in the sentence.

Matplotlib is a library similar to Matlab's plotting library used for plotting various types of pictorial representations.

A class named **SentimentAnalysis** is created. Also a constructor is created to initialise the list of tweets to be collected.

A function **CleanTweet** is created which takes a single tweet from all the collected tweets at a time and checks the pattern specified and then joins it in the new list.

A function is created to calculate the required percentage and return it in a specific format. For plotting a pictorial representation of the results of the analysed data, we have created a function plot PieChart which plots a pie chart according to the number of % of results in a specific category.

A function **SearchForTweet** is created which firstly contains the ConsumerKey, ConsumerSecret, AccessToken and AccessTokenSecret keys which are received by a user when applied successfully for becoming a

Twitter developer. These keys are used to get access to the tweets which otherwise is restricted by Twitter.

OAuthHandler inbuilt function is used to connect to get the connection.

After connecting, we ask the query to be searched for and the no. of tweets required for analysis. **Tweepy.Cursor** inbuilt function is used to get the tweets in which "-filter:retweet " is used to disconsider the retweets as they would create a wrong analysis.

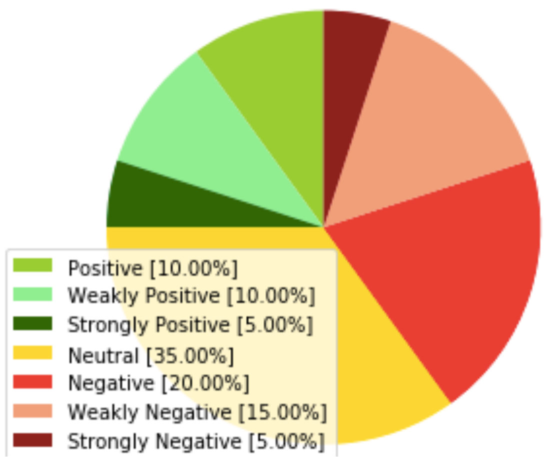
For each tweet, we clean the tweet using the CleanTweet function and then append it to an new list. We find the polarity for each tweet and then pass in a for loop to check the category in which it fits. The categories available are positive, weakly positive, strongly positive, negative, weakly negative, strongly negative and neutral. Each tweet is appended to one of the list of the category it falls into. The range of the polarity is from -1 to 1.

The General report is created stating the overall category of the query searched. The Detailed report gives the percentage of each category. 5 random tweets from each category are printed. For printing all the tweets in all the categories and for file representation, we create a text file and print all the tweets from each category using file handling. Here in the text file we can observe which tweet falls into which category and also the general report and detailed report for the query searched. (See Appendix 2)

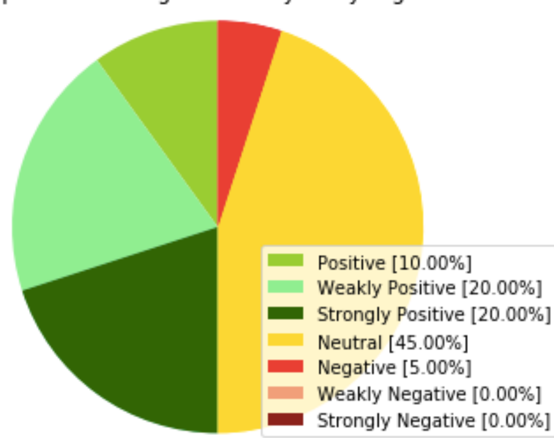
5. RESULT DISCUSSION

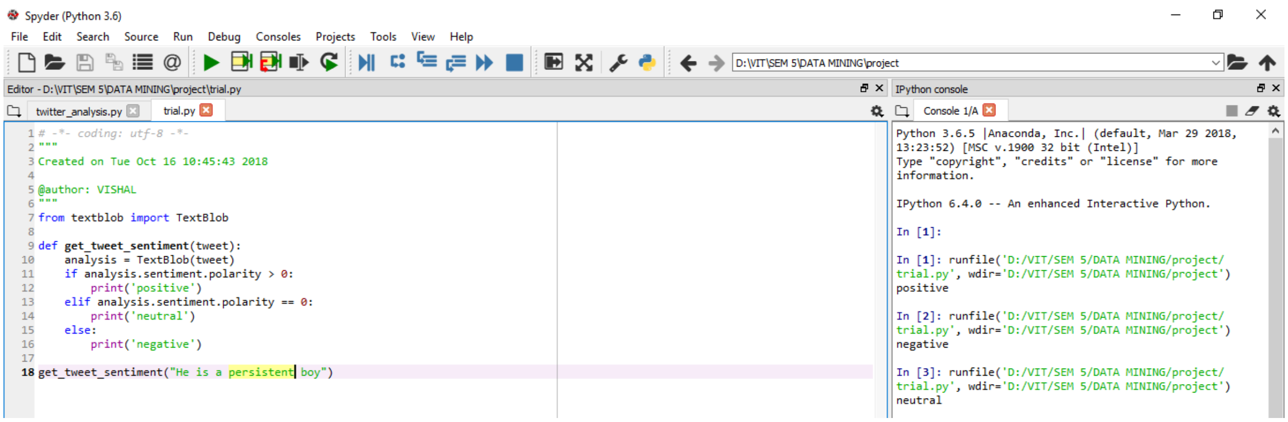
We have taken 2 keywords – India & Donald Trump. We analyzed 20 tweets for each, using pie chart as shown in Appendix 2. We have given 5 Random Tweets for each of the Human Labelling Categories - Positive, Strongly Positive, Weakly Positive, Negative, Weakly Negative, Strongly Negative and Neutral. (Twitter_analysis.py)

How people are reacting on Donald Trump by analyzing 20 Tweets.



How people are reacting on India by analyzing 20 Tweets.





6.APPENDICES

6.1 APPENDIX 1 – SOURCE CODE

Trail.py

```
from textblob import TextBlob

def get_tweet_sentiment(tweet):
    analysis = TextBlob(tweet)
    if analysis.sentiment.polarity > 0:
        print('positive')
    elif analysis.sentiment.polarity == 0:
        print('neutral')
    else:
        print('negative')

get_tweet_sentiment("He wasn't a bad boy")
```

Twitter_analysis.py

```
import tweepy,re
from textblob import TextBlob
import matplotlib.pyplot as plt


class SentimentAnalysis:

    def __init__(self):
        self.tweets = []
        self.tweetText = []

    def cleanTweet(self, tweet):
        # Remove Links, Special Characters etc from tweet
        return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t]) | (\w +:\ / \ / \S +)", " ",
tweet).split())

    # function to calculate percentage
    def percentage(self, part, whole):
        temp = 100 * float(part) / float(whole)
        return format(temp, '.2f')

    def plotPieChart(self, positive, wpositive, spositive, negative, wnegative, snegative,
neutral, searchTerm, noOfSearchTerms):
        labels = ['Positive [' + str(positive) + '%]', 'Weakly Positive [' + str(wpositive) +
'%]','Strongly Positive [' + str(spositive) + '%]', 'Neutral [' + str(neutral) + '%]',
                  'Negative [' + str(negative) + '%]', 'Weakly Negative [' + str(wnegative) + '%]',
'Strongly Negative [' + str(snegative) + '%]']
        sizes = [positive, wpositive, spositive, neutral, negative, wnegative, snegative]
        colors = ['yellowgreen','lightgreen','darkgreen', 'gold', 'red','lightsalmon','darkred']
        patches, texts = plt.pie(sizes, colors=colors, startangle=90)
        plt.legend(patches, labels, loc="best")
        plt.title('How people are reacting on ' + searchTerm + ' by analyzing ' +
str(noOfSearchTerms) + ' Tweets.')
        plt.axis('equal')
        plt.tight_layout()
        plt.show()

    def SearchForTweet(self):
        consumerKey = '20XFqVsP3jLPGghUNwyBLJCp5'
        consumerSecret = 'YAfjnDUfaeWjqbPiT9cvwqk4PbaFsisKAI0MIXrCS1jv9S4NhC'
        accessToken = '3142382286-qfUt9Ynpl3gA1PbSlQoLQHyoWBAf5DkEs8GAvsg'
        accessTokenSecret = 'I8jbKMs6fHGukflwuQFrsWkc8IOakg0NjceNUePVSURhD'

        try:
            auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
```

```
            auth.set_access_token(accessToken, accessTokenSecret)
            api = tweepy.API(auth)
        except:
            print("Error: Authentication Failed")

        # input for term to be searched and how many tweets to search
        searchTerm = input("Enter Keyword/Tag to search about: ")
        NoOfTerms = int(input("Enter how many tweets to search: "))

        # searching for tweets
        self.tweets = tweepy.Cursor(api.search, q=searchTerm+" -filter:retweets", lang =
"en").items(NoOfTerms)
        # creating some variables to store info
        polarity = 0
        positive = 0
        wpositive = 0
        spositive = 0
        negative = 0
        wnegative = 0
        snegative = 0
        neutral = 0

        positives = []
        wpositives = []
        spositives = []
        negatives = []
        wnegatives = []
        snegatives = []
        neutrals = []

        # iterating through tweets fetched
        for tweet in self.tweets:
            #Append to temp so that we can store in csv later. I use encode UTF-8
            cleaned_tweet= self.cleanTweet(tweet.text)
            self.tweetText.append(cleaned_tweet.encode('utf-8'))
            # print (tweet.text.translate(non_bmp_map))    #print tweet's text
            analysis = TextBlob(tweet.text)
            # print(analysis.sentiment)  # print tweet's polarity
            polarity += analysis.sentiment.polarity  # adding up polarities to find the average
later

            if (analysis.sentiment.polarity == 0):  # adding reaction of how people are
reacting to find average later
                neutral += 1
                neutrals.append(cleaned_tweet)
            elif (analysis.sentiment.polarity > 0 and analysis.sentiment.polarity <= 0.3):
```

```
            wpositive += 1
            wpositives.append(cleaned_tweet)
         elif (analysis.sentiment.polarity > 0.3 and analysis.sentiment.polarity <= 0.6):
            positive += 1
            positives.append(cleaned_tweet)
         elif (analysis.sentiment.polarity > 0.6 and analysis.sentiment.polarity <= 1):
            spositive += 1
            spositives.append(cleaned_tweet)
         elif (analysis.sentiment.polarity > -0.3 and analysis.sentiment.polarity <= 0):
            wnegative += 1
            wnegatives.append(cleaned_tweet)
         elif (analysis.sentiment.polarity > -0.6 and analysis.sentiment.polarity <= -0.3):
            negative += 1
            negatives.append(cleaned_tweet)
         elif (analysis.sentiment.polarity > -1 and analysis.sentiment.polarity <= -0.6):
            snegative += 1
            snegatives.append(cleaned_tweet)

      # finding average of how people are reacting
      positive = self.percentage(positive, NoOfTerms)
      wpositive = self.percentage(wpositive, NoOfTerms)
      spositive = self.percentage(spositive, NoOfTerms)
      negative = self.percentage(negative, NoOfTerms)
      wnegative = self.percentage(wnegative, NoOfTerms)
      snegative = self.percentage(snegative, NoOfTerms)
      neutral = self.percentage(neutral, NoOfTerms)

      # finding average reaction
      polarity = polarity / NoOfTerms

      # printing out data
      print("Data Mining Project\n\t\t\t\t-By Manish Cheruku and Vishal Salvankar\n")
      print("How people are reacting on " + searchTerm + " by analyzing " +
str(NoOfTerms) + " tweets.")
      print()
      print("General Report: ")

      if (polarity == 0):
         status= "Neutral"
      elif (polarity > 0 and polarity <= 0.3):
         status= "Weakly Positive"
      elif (polarity > 0.3 and polarity <= 0.6):
         status= "Positive"
      elif (polarity > 0.6 and polarity <= 1):
         status= "Strongly Positive"
      elif (polarity > -0.3 and polarity <= 0):
```

```
            status= "Weakly Negative"
        elif (polarity > -0.6 and polarity <= -0.3):
            status= "Negative"
        elif (polarity > -1 and polarity <= -0.6):
            status= "Strongly Negative"

        print(status)
        print()
        print("Detailed Report: ")
        print(str(positive) + "% people thought it was positive")
        print(str(wpositive) + "% people thought it was weakly positive")
        print(str(spositive) + "% people thought it was strongly positive")
        print(str(negative) + "% people thought it was negative")
        print(str(wnegative) + "% people thought it was weakly negative")
        print(str(snegative) + "% people thought it was strongly negative")
        print(str(neutral) + "% people thought it was neutral")

        self.plotPieChart(positive, wpositive, spositive, negative, wnegative, snegative,
neutral, searchTerm, NoOfTerms)

        print("\n\n5 Random Positive tweets:\n")
        for tweet in positives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Weakly Positive tweets:\n")
        for tweet in wpositives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Strongly Positive tweets:\n")
        for tweet in spositives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Negative tweets:\n")
        for tweet in negatives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Weakly Negative tweets:\n")
        for tweet in wnegatives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Strongly Negative tweets:\n")
        for tweet in snegatives[:5]:
            print('--> ',tweet,'\n')

        print("\n\n5 Random Neutral tweets:\n")
        for tweet in neutrals[:5]:
```

```
        print('--> ',tweet,'\n')

    f = open('analysis.txt','w')
    f.write("Data Mining Project\n\t\t\t\t-By Manish Cheruku and Vishal Salvankar\n")
    f.write("\nHow people are reacting on " + searchTerm + " by analyzing " + 
str(NoOfTerms) + " tweets.")
    f.write("\n\nGeneral Report: \n")
    f.write(status)
    f.write("\n\nDetailed Report: \n")
    f.write(str(positive) + "% people thought it was positive\n")
    f.write(str(wpositive) + "% people thought it was weakly positive\n")
    f.write(str(spositive) + "% people thought it was strongly positive\n")
    f.write(str(negative) + "% people thought it was negative\n")
    f.write(str(wnegative) + "% people thought it was weakly negative\n")
    f.write(str(snegative) + "% people thought it was strongly negative\n")
    f.write(str(neutral) + "% people thought it was neutral\n")
    f.write("\nPositive Tweets: \n")
    for tweet in positives:
        f.write('--> ')
        r= tweet
        for i in range(len(r)):
            try:
                f.write(r[i])
            except:
                fake=1
        f.write('\n')
    f.write("\nWeakly Positive Tweets: \n")
    for tweet in wpositives:
        f.write('--> ')
        r= tweet
        for i in range(len(r)):
            try:
                f.write(r[i])
            except:
                fake=1
        f.write('\n')
    f.write("\nStrongly Positive Tweets: \n")
    for tweet in spositives:
        f.write('--> ')
        r= tweet
        for i in range(len(r)):
            try:
                f.write(r[i])
            except:
                fake=1
        f.write('\n')
```

```
        f.write("\nNegative Tweets: \n")
        for tweet in negatives:
            f.write('--> ')
            r= tweet
            for i in range(len(r)):
                try:
                    f.write(r[i])
                except:
                    fake=1
            f.write('\n')
        f.write("\nWeakly Negative Tweets: \n")
        for tweet in wnegatives:
            f.write('--> ')
            r= tweet
            for i in range(len(r)):
                try:
                    f.write(r[i])
                except:
                    fake=1
            f.write('\n')
        f.write("\nStrongly Negative Tweets: \n")
        for tweet in snegatives:
            f.write('--> ')
            r= tweet
            for i in range(len(r)):
                try:
                    f.write(r[i])
                except:
                    fake=1
            f.write('\n')
        f.write("\nNeutral Tweets: \n")
        for tweet in neutrals:
            f.write('--> ')
            r= tweet
            for i in range(len(r)):
                try:
                    f.write(r[i])
                except:
                    fake=1
            f.write('\n')
        f.close()
def main():
    sa = SentimentAnalysis()
    sa.SearchForTweet()

if __name__== "__main__":
```

main()

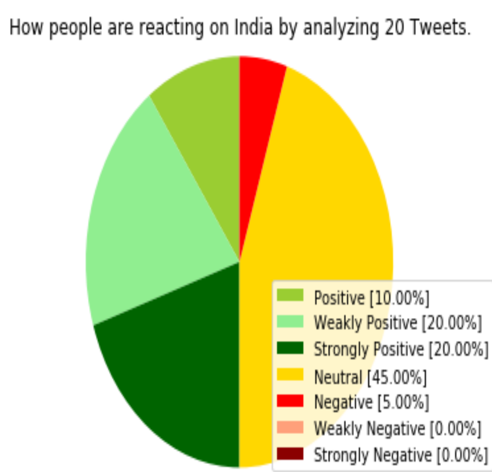
6.2 APPENDIX 2 – SCREENSHOTS / OUTPUTS

Input #1

IPython console

Console 1/A

```
In [8]: runfile('D:/VIT/SEM 5/DATA MINING/project/twitter_analysis.py', wdir='D:/VIT/SEM 5/DATA MINING/project')

Enter Keyword/Tag to search about: India

Enter how many tweets to search: 20
Data Mining Project
                                 -By Manish Cheruku and Vishal Salvankar

How people are reacting on India by analyzing 20 tweets.

General Report:
Weakly Positive

Detailed Report:
10.00% people thought it was positive
20.00% people thought it was weakly positive
20.00% people thought it was strongly positive
5.00% people thought it was negative
0.00% people thought it was weakly negative
0.00% people thought it was strongly negative
45.00% people thought it was neutral
```



```
IPython console
Console 1/A

5 Random Positive tweets:

--> Selectors are not know the importance of dhoni and Raina they played many matches and they played winning knocks for India

--> Sometime I feel Kashmir is the part of world politics agenda since it’s been occupied by India I strongly feel thi https://t.co/OzMOcedfaX


5 Random Weakly Positive tweets:

--> Wonder What happened when Indira Gandhi Rajiv Gandhi Manmohan Singh etc were PM.. Oh wait..They were busy with IN https://t.co/WokkjRv2lt

--> Friends A new breeze of entertainment starts today 1 PM #HelloSagoWithShruti _IN _india _content

--> How much were they paid People of India now can look through modi and his lies BJP will be kicked out in 2019.

--> _Wldlife A Panther caught near Doomdoma Assam #India today morning One person reportedly injured. https://t.co/mBzUuUoWp7


5 Random Strongly Positive tweets:

--> I liked a video https://t.co/gpfnKgMfno Hair Transplant at Dezire Clinic Pune Best Clinic in Maharashtra India for FUE

--> Download GIFkaro India's Largest GIF Library (https://t.co/qhuNsKsfoF) and get a chance to win cash over 800/- https://t.co/rx0UDhigIe

--> Good luck bro make India proud ❤

--> #Partition ws an All Indian deal Good Indians who wanted a united India &amp Rogue Indians who wanted to https://t.co/LEiAk9tl6g


5 Random Negative tweets:

--> Sorry. pls respect the PM. you should have written "Modi ji kahan kahan ghum rahe hain"


5 Random Weakly Negative tweets:


5 Random Strongly Negative tweets:
```

IPython console

Console 1/A

```
5 Random Neutral tweets:

-->  Never mind..does not make any difference..good for India

-->  Which nation India or Pakistan?

-->  Liberals lets start blaming Modi ji and Amit Shah or else let's call this emergency in India even though Modi ji o https://t.co/l7s1lnZeun

-->  _RESEARCH _jp https://t.co/fWiBVDRTKo

-->  Episode 5 of #DearPari 'The Heart Speaks is now out on our website and podcasting apps In this episode we will b https://t.co/pWbCWaGHwq

In [9]:
```

analysis - Notepad

File Edit Format View Help

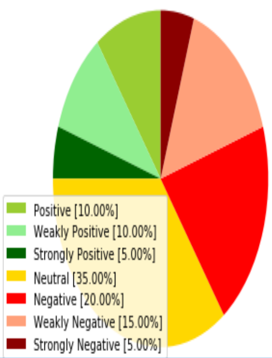
```
Data Mining Project
                                -By Manish Cheruku and Vishal Salvankar

How people are reacting on India by analyzing 20 tweets.

General Report:
Weakly Positive

Detailed Report:
10.00% people thought it was positive
20.00% people thought it was weakly positive
20.00% people thought it was strongly positive
5.00% people thought it was negative
0.00% people thought it was weakly negative
0.00% people thought it was strongly negative
45.00% people thought it was neutral

Positive Tweets:
--> Selectors are not know the importance of dhoni and Raina they played many matches and they played winning knocks for India
--> Sometime I feel Kashmir is the part of world politics agenda since it's been occupied by India I strongly feel thi https://t.co/OzMOcedfaX

Weakly Positive Tweets:
--> Wonder What happened when Indira Gandhi Rajiv Gandhi Manmohan Singh etc were PM.. Oh wait..They were busy with IN https://t.co/WokkjRv2lt
--> Friends A new breeze of entertainment starts today 1 PM #HelloSagoWithShruti _IN _india _content
--> How much were they paid People of India now can look through modi and his lies BJP will be kicked out in 2019.
--> _Wldlife A Panther caught near Doomdoma Assam #India today morning One person reportedly injured. https://t.co/mBzUuUoWp7

Strongly Positive Tweets:
--> I liked a video https://t.co/gpfnKgMfno Hair Transplant at Dezire Clinic Pune Best Clinic in Maharashtra India for FUE
--> Download GIFkaro India's Largest GIF Library (https://t.co/qhuNsKsfoF) and get a chance to win cash over 800/- https://t.co/rx0UDhigIe
--> Good luck bro make India proud
--> #Partition ws an All Indian deal Good Indians who wanted a united India &amp Rogue Indians who wanted to https://t.co/LEiAk9tl6g

Negative Tweets:
--> Sorry. pls respect the PM. you should have written "Modi ji kahan kahan ghum rahe hain"

Weakly Negative Tweets:
```

Activate
Go to Settir

Input #2

```
Enter Keyword/Tag to search about: Donald Trump

Enter how many tweets to search: 20
Data Mining Project
                                -By Manish Cheruku and Vishal Salvankar

How people are reacting on Donald Trump by analyzing 20 tweets.

General Report:
Weakly Negative

Detailed Report:
10.00% people thought it was positive
10.00% people thought it was weakly positive
5.00% people thought it was strongly positive
20.00% people thought it was negative
15.00% people thought it was weakly negative
5.00% people thought it was strongly negative
35.00% people thought it was neutral
```

How people are reacting on Donald Trump by analyzing 20 Tweets.




```
5 Random Positive tweets:

-->  NOW do you get it-? YOU HAVE TO VOTE. This incredibly immoral pretense of a "president is actually threatening th https://t.co/j2WhQYRsmj

-->  How does that fit in with what we were https://t.co/Fg59FjHepi


5 Random Weakly Positive tweets:

-->  Inappropriateness notwithstanding the clarity of thought in this tweet suggest clearly that it's https://t.co/RbnXfmMnr2

-->  Donald Trump I have a two year old boy and I would go to our borders to help protect our countries standards and ri https://t.co/Ty46kY9t2e


5 Random Strongly Positive tweets:

-->  Didn't Donald Trump say "Good On Both Sides? End the violence Vote ALL Republicans out of https://t.co/iP9Oj1LsUP


5 Random Negative tweets:

-->  Donald J Trump needs to be forced to resign He is causing too much hate killing and division in our country.

-->  makes MOSTLY FALSE statements Donald Trump claim that Beto O'Rourke voted to shield gang members https://t.co/idr0ALrkz9

-->  Fake liberals who questioned the audacity of inviting Barak Obama for Republic Day are now mourning the that fact t https://t.co/c2ZfchvyEx

-->  The Dodgers manager dismissed Trump's musings on his bull pen moves during Game 4 of the World Series https://t.co/ryOct0WuEY


5 Random Weakly Negative tweets:

-->  BOOM! Donald Trump The American Bad Ass Featuring Kid Rock STRONG LA.. https://t.co/Om07weSYbU via
```

```
5 Random Weakly Negative tweets:

-->  BOOM! Donald Trump The American Bad Ass Featuring Kid Rock STRONG LA.. https://t.co/Om07weSYbU via

-->  Donald Trump you realy need to take a long look at what your rhetoric played i https://t.co/K8fQgSibeE

-->  US PRESIDENT DONALD TRUMP BOARD GAME HASBRO PARKER BROTHERS POLITICS NEW IN BOX https://t.co/i4SiVFmwwI https://t.co/0Zw7WSrUZS


5 Random Strongly Negative tweets:

-->  The Synagogue in Pittsburgh has been attacked by a terrorist who shouted that he hated all Jews Four people died, https://t.co/rFXKtNvqsP


5 Random Neutral tweets:

-->  Why is it that donald trump is not allowed to talk about baseball just because there was a mass shooting that he's already addressed

-->  Until Donald Trump came to town for Ted Cruz

-->  The Observer view on Donald Trump's vile rhetoric and the US pipe bombs Observer editorial https://t.co/tOAr1g2V8T

-->  "This season of ghouls is animated by the ghost of Roger Ailes who bankrolled by Rupert Murdoch was the mastermi https://t.co/wiNtDxqD38

-->  Chief Keef Doesn't Understand Kanye West's Connection To Donald Trump https://t.co/Eyjxwf9cBf

In [11]:
```

analysis - Notepad

File Edit Format View Help

```
Data Mining Project
                                -By Manish Cheruku and Vishal Salvankar

How people are reacting on Donald Trump by analyzing 20 tweets.

General Report:
Weakly Negative

Detailed Report:
10.00% people thought it was positive
10.00% people thought it was weakly positive
5.00% people thought it was strongly positive
20.00% people thought it was negative
15.00% people thought it was weakly negative
5.00% people thought it was strongly negative
35.00% people thought it was neutral

Positive Tweets:
--> NOW do you get it-? YOU HAVE TO VOTE. This incredibly immoral pretense of a "president is actually threatening th https://t.co/j2WhQYRsmj
--> How does that fit in with what we were https://t.co/Fg59FjHepi

Weakly Positive Tweets:
--> Inappropriateness notwithstanding the clarity of thought in this tweet suggest clearly that it's https://t.co/RbnXfmMnr2
--> Donald Trump I have a two year old boy and I would go to our borders to help protect our countries standards and ri https://t.co/Ty46kY9t2e

Strongly Positive Tweets:
--> Didn't Donald Trump say "Good On Both Sides? End the violence Vote ALL Republicans out of https://t.co/iP9Oj1LsUP

Negative Tweets:
--> Donald J Trump needs to be forced to resign He is causing too much hate killing and division in our country.
--> makes MOSTLY FALSE statements Donald Trump claim that Beto O'Rourke voted to shield gang members https://t.co/idr0ALrkz9
--> Fake liberals who questioned the audacity of inviting Barak Obama for Republic Day are now mourning the that fact t https://t.co/c2ZfchvyEx
--> The Dodgers manager dismissed Trump's musings on his bull pen moves during Game 4 of the World Series https://t.co/ryOct0WuEY

Weakly Negative Tweets:
--> BOOM! Donald Trump The American Bad Ass Featuring Kid Rock STRONG LA.. https://t.co/Om07weSYbU via
--> Donald Trump you realy need to take a long look at what your rhetoric played i https://t.co/K8fQgSibeE
```

7. REFERENCES

[1] Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner and Isabell M. Welpe. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. *In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM)*, 2010.

[2] Theresa Wilson, Janyce Wiebe and Paul Hoffmann. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. *In the Annual Meeting of Association of Computational Linguistics: Human Language Technologies (ACL-HLT)*, 2005.

[3] Peter D. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *In Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 2002.

[4] Samuel Brody and Nicholas Diakopoulus. Using Word Lengthening to Detect Sentiment in Microblogs. *In Proceedings of Empirical Methods on Natural Language Processing (EMNLP)*, 2011.

[5] Alexander Pak and Patrick Paroubek. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *In Proceedings of international conference on Language Resources and Evaluation (LREC)*, 2010.

[6] Luciano Barbosa and Junlan Feng. Robust Sentiment Detection on Twitter from Biased and Noisy Data. *In Proceedings of the international conference on Computational Linguistics (COLING)*, 2010.

[7] fthymios Kouloumpis, Theresa Wilson and Johanna Moore. Twitter Sentiment Analysis: The Good the Bad and the OMG! *In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.

[8] Alec Go, Richa Bhayani and Lei Huang. Twitter Sentiment Classification using Distant Supervision. *Project Technical Report, Stanford University*, 2009.

[9] Multi Perspective Question Answering (MPQA) Online Lexicon <http://www.cs.pitt.edu/mpqa/subj_lexicon.html>

[10]Stefano Baccianella, Andrea Esuli, Fabrizio Sebastiani. SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. *In Proceedings of international conference on Language Resources and Evaluation (LREC)*, 2010.

[11]Peter D. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *In Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 2002.

[12]Rudy Prabowo and Mike Thelwall. Sentiment Analysis: A Combined Approach. Journal of Infometrics, Volume 3, Issue 2, April 2009, Pages 143-157, 2009.

[13]Hatzivassiloglou, V., & McKeown, K.R.. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL,* 2009.

[14]https://textblob.readthedocs.io/en/dev/quickstart.html#sentiment-analysis

[15]https://textblob.readthedocs.io/en/dev/_modules/textblob/en/sentiments.html