TWITTER SENTIMENT ANALYSIS USING PYTHON

Abstract*: In this paper, we accomplished a project on Twitter tweets. Using Twitter API, we collected few tweets and performed sentiment analysis. In this sentiment analysis, we classified tweets as positive, negative, and neutral based on Textblob lexicon of words. Thus, we collected public tweets on great personalities like Donald Trump, Barack Obama, Narendra Modi, and Hillary Clinton. We analyzed their tweets and could calculate most frequently used words in the tweets made by users on the great personalities. Moreover, we created a word cloud with the frequently used words. Finally, we classified number of tweets which were positive, negative, and neutral on the great personalities.*

Keywords*: **Twitter, API, Tweets, and Textblob.***

Introduction:

Twitter is an online news and social networking service where users post and interact with messages, "tweets," restricted to 140 characters [1]. Registered users can post tweets, but those who are unregistered can only read them [1].

The REST APIs provide programmatic access to read and write Twitter data [2]. Create a new Tweet, read user profile and follower data, and more [2]. The REST API identifies Twitter applications and users using OAuth; responses are in JSON format [2].

Sentiment Analysis is the process of 'computationally' determining whether a piece of writing is positive, negative or neutral [3]. It's also known as **opinion mining**, deriving the opinion or attitude of a speaker [3].

Why sentiment analysis?

- **Business:** In marketing field companies use it to develop their strategies, to understand customers' feelings towards products or brand, how people respond to their campaigns or product launches and why consumers don't buy some products [3].
- **Politics:** In political field, it is used to keep track of political view, to detect consistency and inconsistency between statements and actions at the government level. It can be used to predict election results as well! [3].
- **Public Actions:** Sentiment analysis also is used to monitor and analyze social phenomena, for the spotting of potentially dangerous situations and determining the general mood of the blogosphere [3].

Methodology:

Installation

Jupyter Notebook: We used this IDE to compile code written in Python. It is a beautiful notebook with comments, headings, code, graphs, and results.

Tweepy: tweepy is the python client for the official Twitter API [3].

Textblob: textblob is the python library for processing textual data [3].

Authentication

To fetch tweets through Twitter API, one needs to register an App through their twitter account. Follow these steps for the same:

- Open this link and click the button: 'Create New App'
- Fill the application details. You can leave the callback url field empty.
- Once the app is created, you will be redirected to the app page.
- Open the 'Keys and Access Tokens' tab.
- Copy 'Consumer Key', 'Consumer Secret', 'Access token' and 'Access Token Secret'.

Data Acquisition and Analysis:

We followed these 3 major steps in our program:

- Authorize twitter API client.
- Make a GET request to Twitter API to fetch tweets for a query.
- Parse the tweets. Classify each tweet as positive, negative or neutral.

Now, let us try to understand the above piece of code:

- First, we create a **TwitterClient** class. This class contains all the methods to interact with Twitter API and parsing tweets. We use **__init__** function to handle the authentication of API client [3].
- In **get_tweets** function, we use:

fetched_tweets = self.api.search(q = query, count = count)

to call the Twitter API to fetch tweets.

- In **get_tweet_sentiment** we use textblob module.

analysis = TextBlob(self.clean_tweet(tweet))

TextBlob is a high-level library built over top of NLTK library. First, we call **clean_tweet** method to remove links, special characters, etc. from the tweet using some simple regex [3].

Then, as we pass **tweet** to create a **TextBlob** object, following processing is done over text by textblob library:

- Tokenize the tweet, i.e. split words from body of text.
- Remove stopwords from the tokens (stopwords are the commonly used words which are irrelevant in text analysis like I, am, you, are, etc.) [3].
- Do POS (part of speech) tagging of the tokens and select only significant features/tokens like adjectives, adverbs, etc. [3].
- Pass the tokens to a **sentiment classifier** which classifies the tweet sentiment as positive, negative or neutral by assigning it a polarity between -1.0 to 1.0 [3].

Here is how **sentiment classifier** is created:

- **TextBlob** uses a Movies Reviews dataset in which reviews have already been labelled as positive or negative [3].
- Positive and negative features are extracted from each positive and negative review respectively [3].
- Training data now consists of labelled positive and negative features. This data is trained on a Naive Bayes Classifier [3].

Then, we use **sentiment.polarity** method of **TextBlob** class to get the polarity of tweet between -1 to 1.
Then, we classify polarity as:

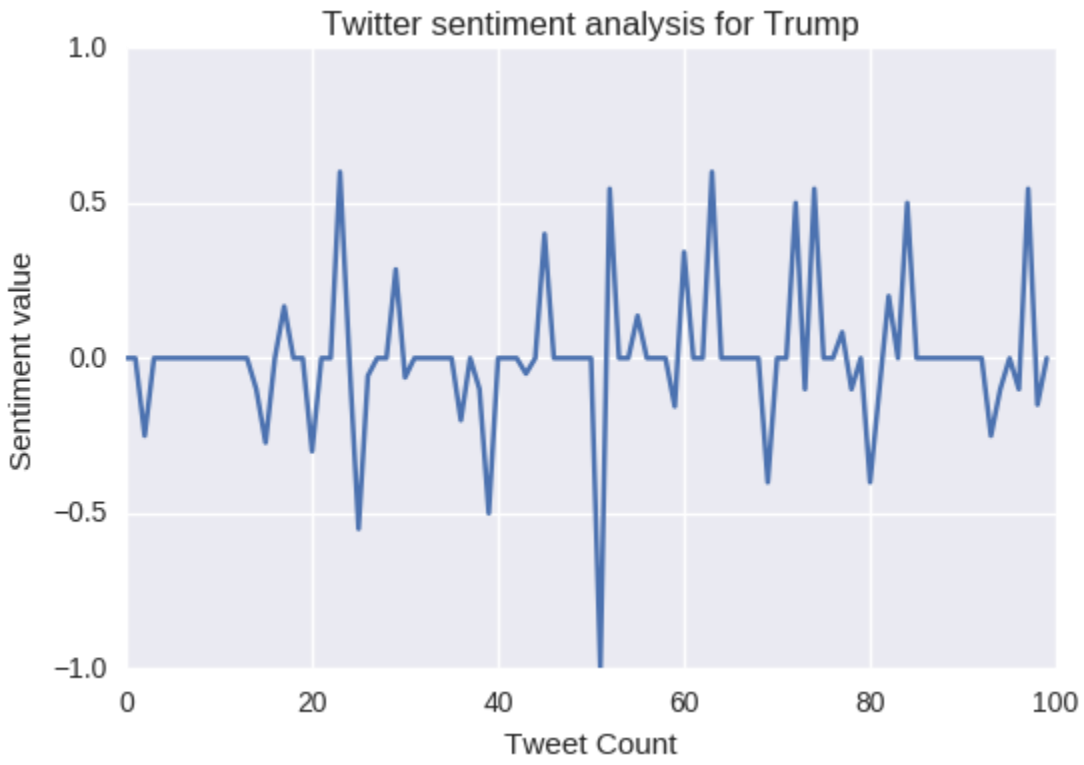
```
if analysis.sentiment.polarity > 0:
    return 'positive'
elif analysis.sentiment.polarity == 0:
    return 'neutral'
else:
    return 'negative'
```

- Finally, parsed tweets are returned. Then, we can do various type of statistical analysis on the tweets. For example, in above program, we tried to find the percentage of positive, negative and neutral tweets about a query [3].

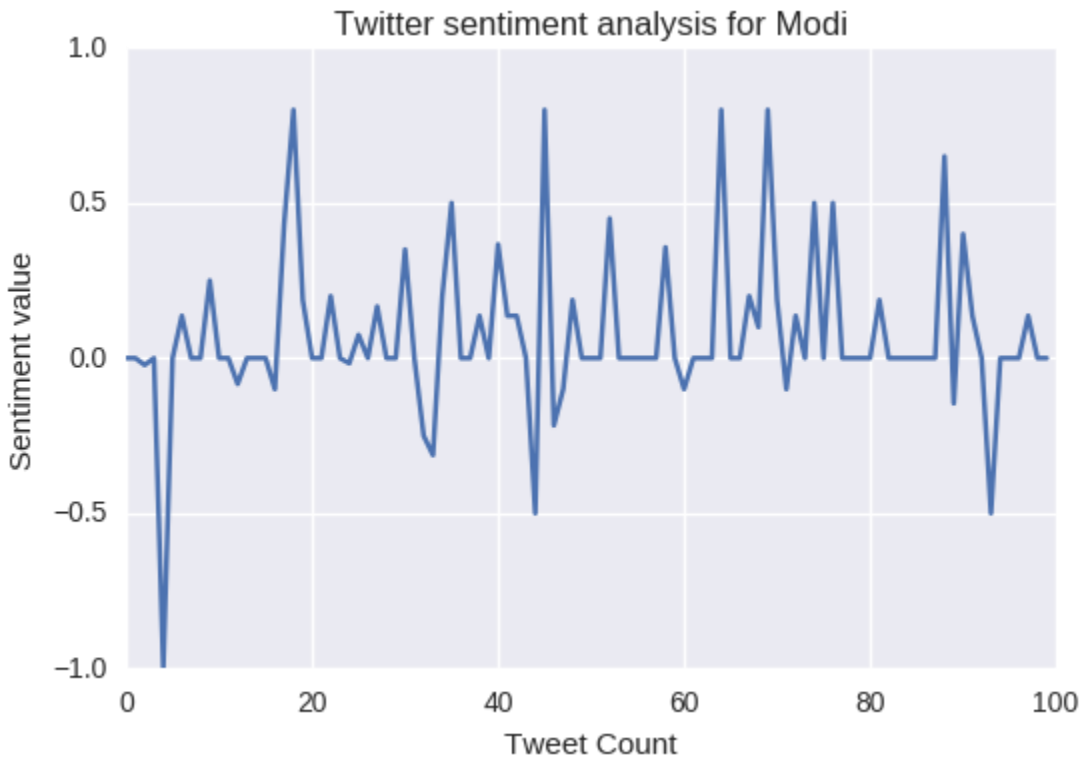
Demonstration:

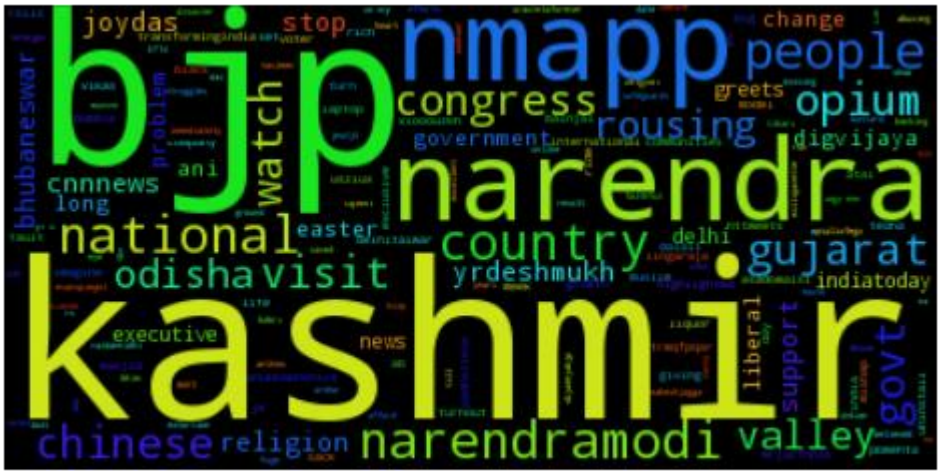
[Course-Project-code] (click to see source code in html format)

Results:

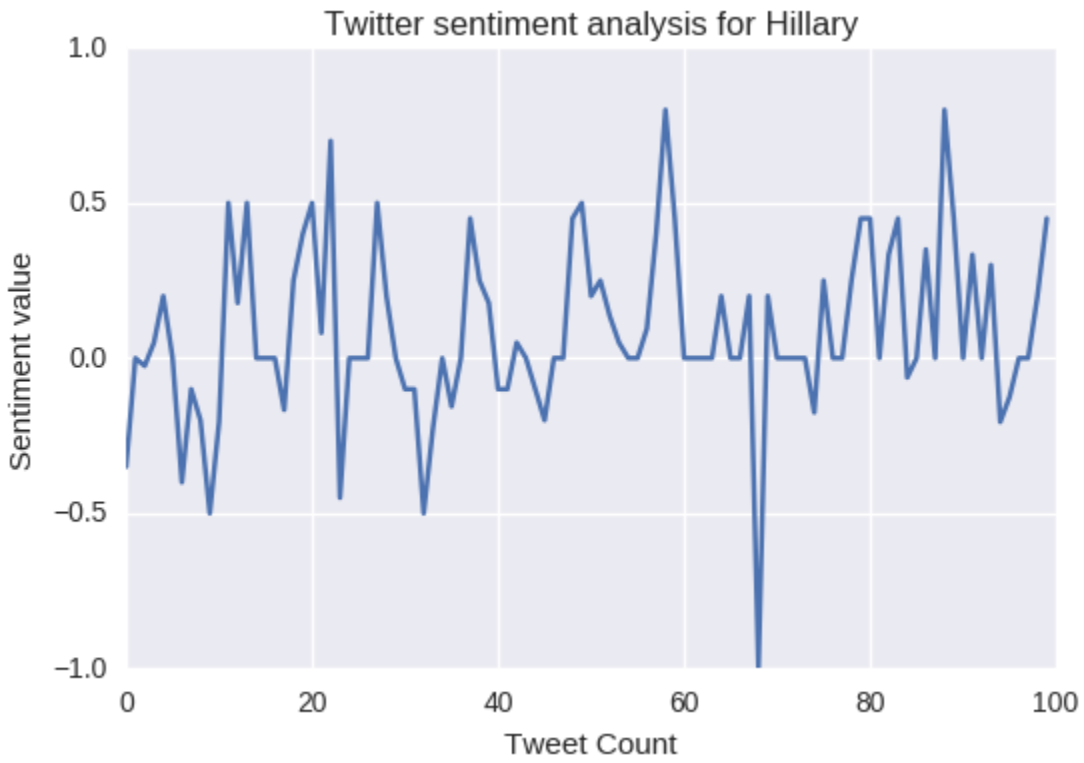


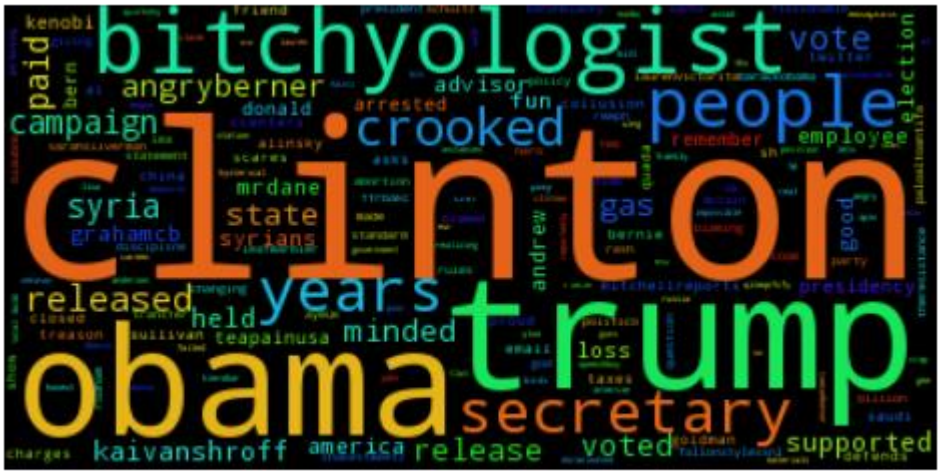
Word Cloud for frequently used words in tweets for Trump



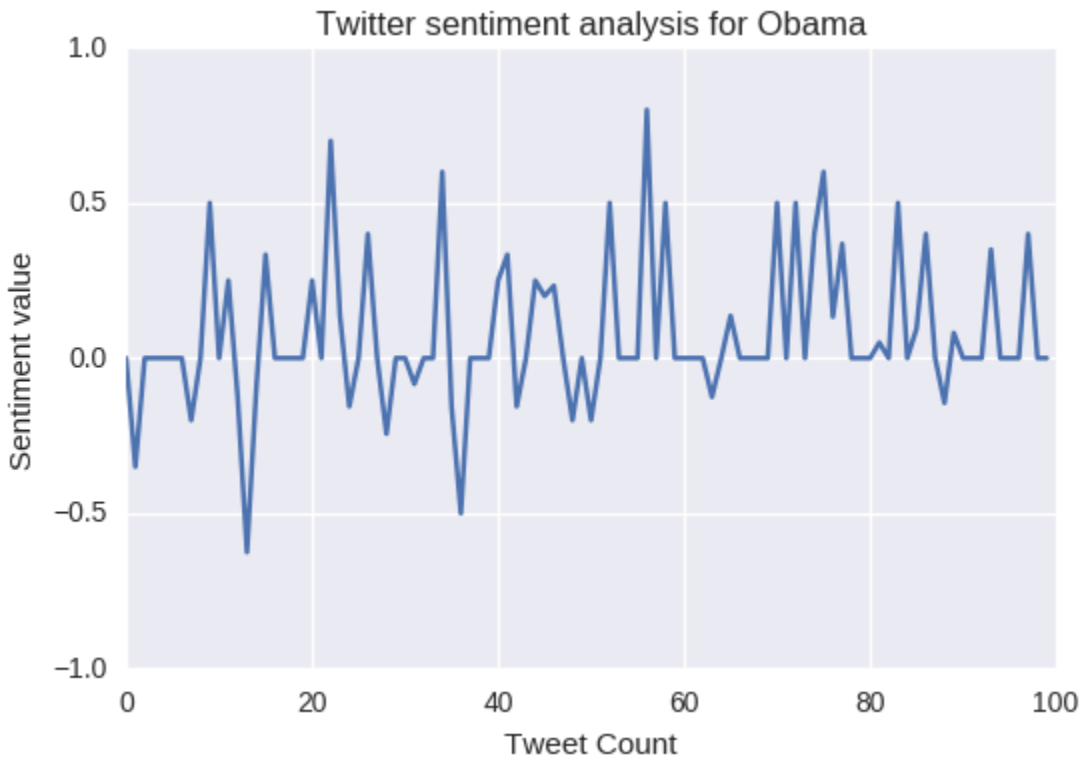


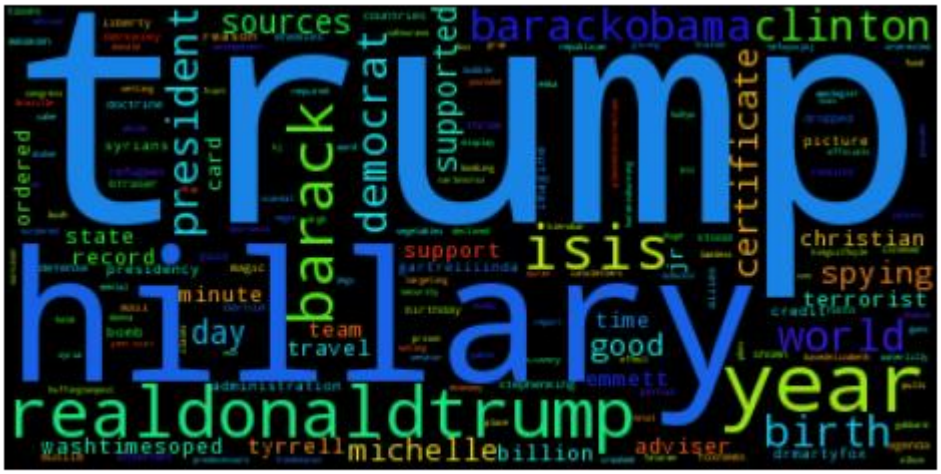
Word Cloud for frequently used words in tweets for Modi





Word Cloud for frequently used words in tweets for Hillary





Word Cloud for frequently used words in tweets for Obama

Ratings:

16.0 % positive, 25.33 % negative, 58.67 % neutral tweets.

Total sentiment value for Trump is 0.15

40.0 % positive, 18.18 % negative, 41.82 % neutral tweets

Total sentiment value for Modi is 7.16

34.85 % positive, 31.82 % negative, 33.33 % neutral tweets.

Total sentiment value for Hillary is 8.44

31.25 % positive, 17.5 % negative, 51.25 % neutral tweets.

Total sentiment value for Obama is 7.39

Conclusion:

Thus, we accomplished our project with sentiment analysis on collected tweets for search_query = = 'Trump', 'Modi', 'Hillary', and 'Obama'. Finally, created a word cloud based on most frequently used words from tweets.

Contributions:

Kartheek Midde – 50%

Sai Nikhil Bheemanathini – 50%

References:

[1] "Twitter," *Wikipedia*. 23-Apr-2017.
[2] "REST APIs — Twitter Developers." [Online]. Available: https://dev.twitter.com/rest/public. [Accessed: 23-Apr-2017].
[3] "Twitter Sentiment Analysis using Python," *GeeksforGeeks*, 24-Jan-2017. .