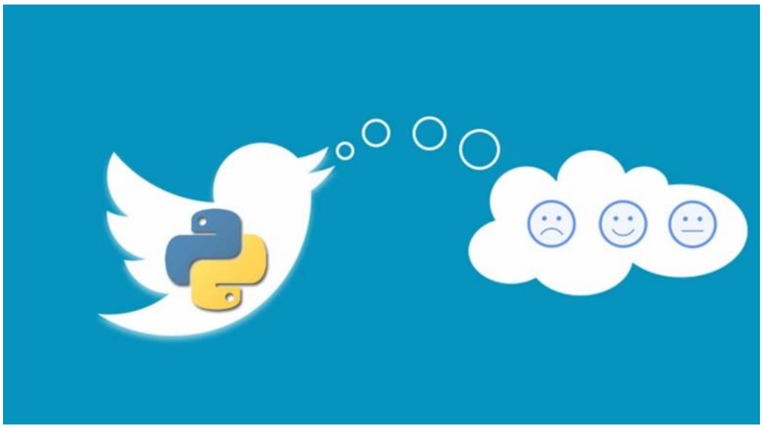


TWITTER SENTIMENT ANALYSIS USING PYTHON

ASHISH ALICHEN
2K19CSUN01127
B.TECH CSE 7A

Abstract

Sentiment Analysis is the process of ‘computationally’ determining whether a piece of writing is positive, negative or neutral. It’s also known as opinion mining, deriving the opinion or attitude of a speaker. It can also be used to compare any queries or determine overall sentiment based on the public opinion on social media.

- Business: In marketing field companies use it to develop their strategies, to understand customers’ feelings towards products or brand.
- Politics: 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!
- Public Actions: Sentiment analysis also is used to monitor and analyze social phenomena

We take user input for a query, fetch tweets for that keyword/s and tag each of the tweet as positive or negative to estimate the general sentiment.

Methodology

Libraries:

- **Tweepy**

It is used to authenticate user for accessing the Twitter API We create an instance of the OAuthHandler type and pass the parameters:

a. Consumer Key
b. Consumer Secret
c. Access Token
d. Access Token Secret

- **TextBlob**

TextBlob is a Python library for processing textual data. First we call clean_tweet method to remove links, special characters, etc. Then, as we pass tweet to create a TextBlob object. Further, the TextBlob library

a. tokenizes the tweet
b. Remove stopwords from the tokens
c. assigns it a polarity value ranging from -1.0 to 1.0

Installation:

- **Tweepy:** tweepy is the python client for the official Twitter API.
  Install it using following pip command:
  ```
  pip install tweepy
  ```
- **TextBlob:** textblob is the python library for processing textual data.
  Install it using following pip command:
  ```
  pip install textblob
  ```
  Also, we need to install some NLTK corpora using following command:
  ```
  python -m textblob.download_corpora
  ```
  (Corpora is nothing but a large and structured set of texts.)

Authentication:

In order 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'.

We follow these 3 major steps in our program:

- Authorize twitter API client.
- Make a GET request to Twitter API to fetch tweets for a particular query.

- Parse the tweets. Classify each tweet as positive, negative or neutral.

Working of the code:

- First of all, 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.
- 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 actually 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.

  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.)
  - Do POS(part of speech) tagging of the tokens and select only significant features/tokens like adjectives, adverbs, etc.
  - 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 .
- Here is how **sentiment classifier** is created:

  - **TextBlob** uses a Movies Reviews dataset in which reviews have already been labelled as positive or negative.

    - Positive and negative features are extracted from each positive and negative review respectively.
    - Training data now consists of labelled positive and negative features. This data is trained on a Naive Bayes Classifier.

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.

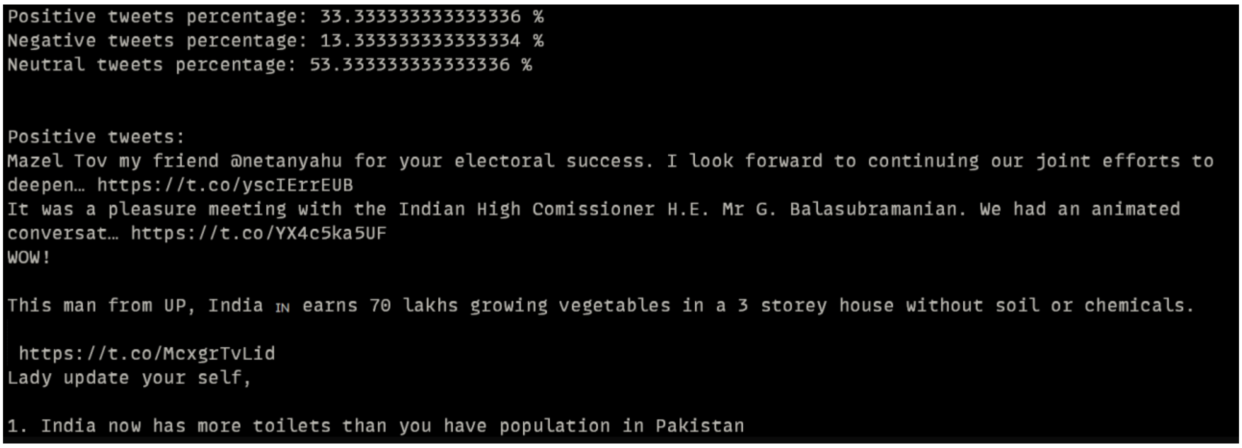
Code

```
import re

import tweepy

from tweepy import OAuthHandler

from textblob import TextBlob
```

```
class TwitterClient(object):
    '''
    Generic Twitter Class for sentiment analysis.
    '''

    def __init__(self):
        '''
        Class constructor or initialization method.
        '''
        # keys and tokens from the Twitter Dev Console
        consumer_key = '6kQ0i7tHfnDpn060OqZOBleie'
        consumer_secret =
'vvfHQmD6G0gSifbycqjllLgmhA51HscQ89zPRpET8tZx4brlk7'
        access_token = '1416842455178043394-
ZkvEjSyla1nphIc04mTDaYCHXpw1Dz'
        access_token_secret =
'9TyC4ev6Ko5EM6ZflBbkKVqZsU0su6UTF2YFsXtFUajDj'

        # attempt authentication
        try:
            # create OAuthHandler object
            self.auth = OAuthHandler(consumer_key,
consumer_secret)
            # set access token and secret
            self.auth.set_access_token(access_token,
access_token_secret)
            # create tweepy API object to fetch tweets
            self.api = tweepy.API(self.auth)
        except:
            print("Error: Authentication Failed")

    def clean_tweet(self, tweet):
        '''
        Utility function to clean tweet text by removing links,
special characters
        using simple regex statements.
        '''
        return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z
\t])|(\w+:\/\/\S+)", " ", tweet).split())

    def get_tweet_sentiment(self, tweet):
        '''
        Utility function to classify sentiment of passed tweet
        using textblob's sentiment method
        '''
        # create TextBlob object of passed tweet text
        analysis = TextBlob(self.clean_tweet(tweet))
        # set sentiment
        if analysis.sentiment.polarity > 0:
            return 'positive'
        elif analysis.sentiment.polarity == 0:
            return 'neutral'
        else:
            return 'negative'

    def get_tweets(self, query, count=100):
        '''
        Main function to fetch tweets and parse them.
        '''
        # empty list to store parsed tweets
        tweets = []

        try:
            # call twitter api to fetch tweets
            fetched_tweets = self.api.search_tweets(q=query, count=count, result_type='popular')

            # parsing tweets one by one
            for tweet in fetched_tweets:
                # empty dictionary to store required params of a tweet
                parsed_tweet = {}

                # saving text of tweet
                parsed_tweet['text'] = tweet.text
                # saving sentiment of tweet
                parsed_tweet['sentiment'] =
self.get_tweet_sentiment(
                    tweet.text)

                # appending parsed tweet to tweets list
                if tweet.retweet_count > 0:
                    # if tweet has retweets, ensure that it is
appended only once
                    if parsed_tweet not in tweets:
                        tweets.append(parsed_tweet)
                else:
                    tweets.append(parsed_tweet)

            # return parsed tweets
            return tweets

        except tweepy.TweepyException as e:
            # print error (if any)
            print("Error : " + str(e))


def main():
    # creating object of TwitterClient Class
    api = TwitterClient()

    ####################### EDIT HERE
#######################

    # q = input("Enter the query keywords: ")

    tweets = api.get_tweets(query='india', count=200)


    # picking positive tweets from tweets

    ptweets = [tweet for tweet in tweets if tweet['sentiment']
== 'positive']

    # percentage of positive tweets

    print("Positive tweets percentage: {} %".format(

        100 * len(ptweets) / len(tweets)))

    # picking negative tweets from tweets

    ntweets = [tweet for tweet in tweets if tweet['sentiment']
== 'negative']

    # percentage of negative tweets

    print("Negative tweets percentage: {} %".format(

        100 * len(ntweets) / len(tweets)))

    # percentage of neutral tweets

    print("Neutral tweets percentage: {} % \

        ".format(100 * (len(tweets) - (len(ntweets) +
len(ptweets))) / len(tweets)))


    # printing first 5 positive tweets

    print("\n\nPositive tweets:")

    for tweet in ptweets[:10]:
        print(tweet['text'])

    # printing first 5 negative tweets
    print("\n\nNegative tweets:")
    for tweet in ntweets[:10]:
        print(tweet['text'])

    print('\n')



if __name__ == "__main__":
    # calling main function
    main()
```

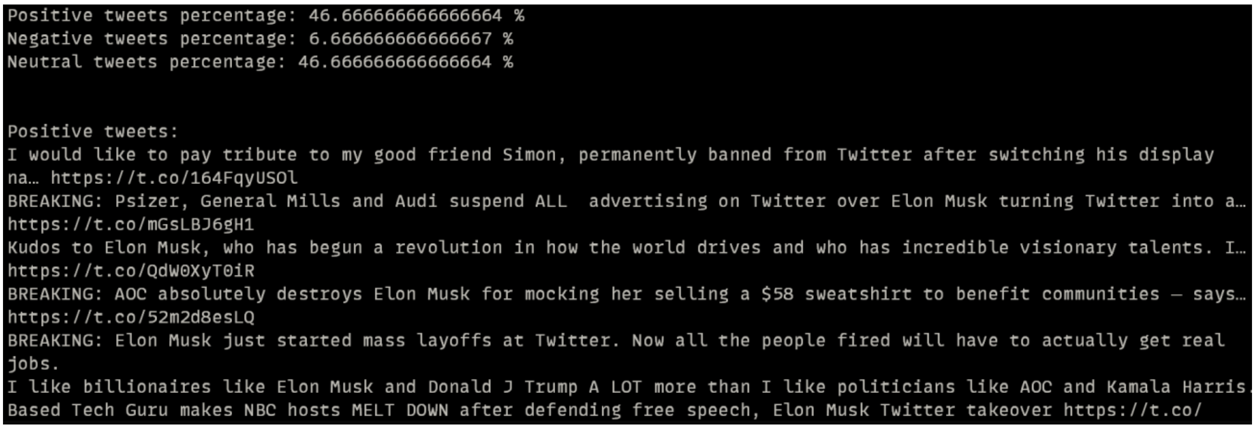
Output

Query : ‘India’



```
Positive tweets percentage: 33.333333333333336 %
Negative tweets percentage: 13.333333333333334 %
Neutral tweets percentage: 53.333333333333336 %


Positive tweets:
Mazel Tov my friend @netanyahu for your electoral success. I look forward to continuing our joint efforts to
deepen… https://t.co/yscIErrEUB
It was a pleasure meeting with the Indian High Comissioner H.E. Mr G. Balasubramanian. We had an animated
conversat… https://t.co/YX4c5ka5UF
WOW!

This man from UP, India IN earns 70 lakhs growing vegetables in a 3 storey house without soil or chemicals.

 https://t.co/McxgrTvLid
Lady update your self,

1. India now has more toilets than you have population in Pakistan
```

Query: ‘Elon musk’



```
Positive tweets percentage: 46.666666666666664 %
Negative tweets percentage: 6.666666666666667 %
Neutral tweets percentage: 46.666666666666664 %


Positive tweets:
I would like to pay tribute to my good friend Simon, permanently banned from Twitter after switching his display
na… https://t.co/164FqyUSOl
BREAKING: Psizer, General Mills and Audi suspend ALL  advertising on Twitter over Elon Musk turning Twitter into a…
https://t.co/mGsLBJ6gH1
Kudos to Elon Musk, who has begun a revolution in how the world drives and who has incredible visionary talents. I…
https://t.co/QdW0XyT0iR
BREAKING: AOC absolutely destroys Elon Musk for mocking her selling a $58 sweatshirt to benefit communities — says…
https://t.co/52m2d8esLQ
BREAKING: Elon Musk just started mass layoffs at Twitter. Now all the people fired will have to actually get real
jobs.
I like billionaires like Elon Musk and Donald J Trump A LOT more than I like politicians like AOC and Kamala Harris.
Based Tech Guru makes NBC hosts MELT DOWN after defending free speech, Elon Musk Twitter takeover https://t.co/
```