

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in

Information Technology

By

RAMYA U
(Reg. No. 37120057)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI-60011.TAMILNADU.

JANUARY 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" Grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119.

Phone: 044 - 2450 3150 / 51 / 52 / 54 / 55 Fax: 044 - 2450 2344

www.sathyabama.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Professional Training Report is the bonafide work of RAMYA.U (Reg. No. 37120057) who underwent the professional training in "TWITTER SENTIMENT ANALYSIS" under our supervision from July 2019 to September 2019.

Internal Guide

Dr. P. AJITHA, Ph.D.,

Head of the Department

Dr. R. SUBHASHINI, M.E., Ph.D.,

Submitted for Viva voce Examination held

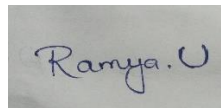
on _____

Internal Examiner

External Examiner

DECLARATION

I, RAMYA.U (Reg. No. 37120057) hereby declare that the Professional Training Report on “**TWITTER SENTIMENT ANALYSIS**” done by me under the guidance of Dr. P. AJITHA Ph.D., at Sathyabama Institute Of Science And Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of technology degree in Information Technology.

A rectangular box containing a handwritten signature in blue ink that reads "Ramya.U".

DATE:

SIGNATURE OF THE CANDIDATE

PLACE:

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D, Dean**, School of Computing , **Dr.SUBHASHINI M.E., Ph.D. ,** Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express y sincere and deep sense of gratitude to my Project Guide **Dr. P. AJITHA, Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

This certificate is computer generated and can be verified by scanning the QR code given below. This will display the certificate from the NPTEL repository, <https://npTEL.ac.in/noc/>

Roll No: NPTEL20CS48S12390036

To
RAMYA U
204, MANTHAI MANNA VELAR STREET
NEIVELY SOUTH
THANJAVUR
TAMIL NADU - 614628
PH. NO :7402350536



Score	Type of Certificate
>=90	Elite+Gold
75-89	Elite+Silver
>=60	Elite
40-59	Successfully Completed
<40	No Certificate

No. of credits recommended by NPTEL:2

An additional 1 credit may be awarded if the University deems it fit, based on the actual student effort involved.



Elite NPTEL Online Certification

(Funded by the Ministry of HRD, Govt. of India)



This certificate is awarded to

RAMYA U

for successfully completing the course

Cloud Computing and Distributed Systems

with a consolidated score of **75** %

Online Assignments	24.50/25	Proctored Exam	50.25/75
--------------------	----------	----------------	----------

Devendra Jalihal

Prof. Devendra Jalihal
Chairman
Centre for Continuing Education, IITM

Total number of candidates certified in this course: 77

Jan-Mar 2020
(8 week course)

Prof. Andrew Thangaraj

Prof. Andrew Thangaraj
NPTEL Coordinator
IIT Madras



Indian Institute of Technology Madras



Roll No: NPTEL20CS48S12390036

To validate and check scores: <https://npTEL.ac.in/noc>

ABSTRACT

TWITTER SENTIMENT ANALYSIS

This project addresses the problem of sentiment analysis in twitter during the current situation covid-19; that is classifying tweets according to the sentiment expressed in them: positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million registered users [24] - out of which 100 million are active users and half of them log on twitter on a daily basis - generating nearly 250 million tweets per day [20]. Due to this large amount of usage we hope to achieve a reflection of public sentiment by analysing the sentiments expressed in the tweets. Analysing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	vi
	LIST OF FIGURES	ix
	LIST OF ABBREVIATIONS	x
1	INTRODUCTION 1.1 Project Outline 1.2 Introduction	1
2	SYSTEM REQUIREMENTS 2.1 Software Requirements 2.2 Hardware Requirements	3
3	MODULES AND FUNCTIONS 3.1 Modules 3.2 Functions	4
4	MATERIALS AND METHODS 4.1 Description 4.2 Take Input 4.3 Encode 4.4 Generate QR Code 4.5 Decode and Display	6

5	RESULT	9
	5.1 Output	
6	CONCLUSION	11
	REFERENCE	12
	APPENDIX	13
	A. SOURCE CODE	
	B. SCREEN SHOTS	

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
4.1.1	Create An Application	6
4.2.1	Registering The Application	7
5.1.1	Positive Tweets	9
5.1.2	Tweets Explanation	9
5.1.3	Negative Tweets	10
5.1.4	Tweets Explanation	10

LIST OF ABBREVIATION

IDLE	-	INTEGRATED DEVELOPMENT ENVIRONMENT
API	-	APPLICATION PROGRAMMING INTERFACE
NLP	-	NATURAL LANGUAGE PROCESSING

CHAPTER 1

INTRODUCTION

1.1 PROJECT OUTLINE

TITLE OF THE PROJECT: - TWITTER SENTIMENT ANALYSIS

1.2 INTRODUCTION

As internet is growing bigger, its horizons are becoming wider. Social Media and Micro blogging platforms like Facebook, Twitter, Tumblr dominate in spreading encapsulated news and trending topics across the globe at a rapid pace. A topic becomes trending if more and more users are contributing their opinion and judgments, thereby making it a valuable source of online perception. These topics generally intended to spread awareness or to promote public figures, political campaigns during elections, product endorsements and entertainment like movies, award shows. Large organizations and firms take advantage of people 's feedback to improve their products and services which further help in enhancing marketing strategies. One such example can be leaking the pictures of upcoming iPhone to create a hype to extract people's emotions and market the product before its release. Thus, there is a huge potential of discovering and analyzing interesting patterns from the infinite social media data for business-driven applications. Sentiment analysis is the prediction of emotions in a word, sentence or corpus of documents. It is intended to serve as an application to understand the attitudes, opinions and emotions expressed within an online mention. The intention is to gain an overview of the wider public opinion behind certain topics. Precisely,it is a paradigm of

categorizing conversations into positive, negative or neutral labels. Many people use social media sites for networking with other people and to stay up-to-date with news and current events. These sites (Twitter, Facebook, Instagram, google+) offer a platform to people to voice their opinion. For example, people quickly post their reviews online as soon as they watch a movie and then start a series of comments to discuss about the acting skills depicted in the movie. This kind of information forms a basis for people to evaluate, rate about the performance of not only any movie but about other products and to know about whether it will be a success or not. This type of vast information on these sites can be used for marketing and social studies. Therefore, sentiment analysis has wide applications and includes emotion mining, polarity, classification and influence analysis. Twitter is an online networking site driven by tweets which are 140 character limited messages. Thus, the character limit enforces the use of hashtags for text classification. Currently around 6500 tweets are published per second, which results in approximately 561.6 million tweets per day. The Twitter sentiment analysis involves the use of natural language processing to extract, identify and characterize the sentiment content. Sentiment Analysis is often carried out at two levels 1) coarse level and 2) fine level. In coarse level, the analysis of entire documents is done while in fine level, the analysis of attributes is done. The sentiments present in the text are of two types: Direct and Comparative. In comparative sentiments, the comparison of objects in the same sentence is involved while in direct sentiments, objects are independent of one another in the same sentence. However, they tend to be highly unstructured and non-grammatical. It gets difficult to interpret their meaning moreover extensive usage of slang words, acronyms and out of vocabulary words are quite common while tweeting online.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 SOFTWARE REQUIREMENTS :

Operating System : Windows 8

Language : python

software used : python IDLE

2.2 HARDWARE REQUIREMENTS :

Minimum 1GB RAM.

Minimum 50MB hard disk.

Better performance with 2.10ghz and above cache memory

CHAPTER 3

MODULES AND FUNCTIONS

3.1 MODULES:

Following python module have been used in the program:

- tweepy
- datetime
- TextBlob
- Matplotlib

3.2 FUNCTIONS:

The functions that are used are as follows:

- Description
- Registering App
- Accessing Data
- Storing Data
- Preparing Data

Tweepy:

Tweepy is open-sourced, hosted on GitHub and enables Python to communicate with Twitter platform and use its API. Tweepy supports accessing Twitter via Basic Authentication and the newer method, OAuth. Twitter has stopped accepting Basic Authentication so OAuth is now the only way to use the Twitter API. The main difference between Basic and OAuth authentication are the consumer and access keys. With Basic Authentication, it was possible to provide a username and password and

access the API, but since 2010 when the Twitter started requiring OAuth, the process is a bit more complicated. An app has to be created at dev.twitter.com. OAuth is a bit more complicated initially than Basic Auth, since it requires more effort, but the benefits it offers are very lucrative: Tweets can be customized to have a string which identifies the app which was used. It doesn't reveal user password, making it more secure. It's easier to manage the permissions, for example a set of tokens and keys can be generated that only allows reading from the timelines, so in case someone obtains those credentials, he/she won't be able to write or send direct messages, minimizing the risk. The application doesn't rely on a password, so even if the user changes it, the application will still work.

TextBlob

TextBlob is a Python (2 and 3) library for processing textual data. 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.

Matplotlib

matplotlib.pyplot is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc. In matplotlib.pyplot various states are preserved across function calls, so that it keeps track of things like the current figure and plotting area, and the plotting functions are directed to the current axes.

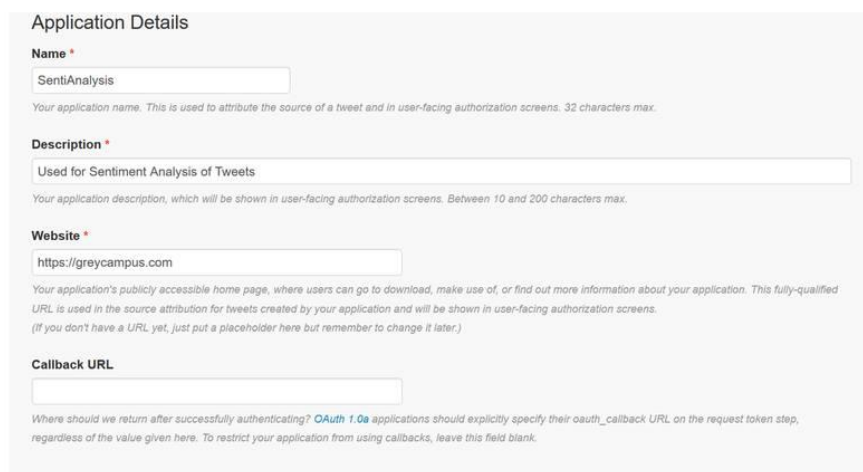
CHAPTER 4

MATERIALS AND METHODS

4.1 Description:

To be able to access Twitter data programmatically we need to create and register an app on twitter developers website for authentication and thereafter we can access data by using Twitter API.

Create an application



The screenshot shows the 'Create an application' form on the Twitter developer website. The form is titled 'Application Details' and contains four main sections: 'Name', 'Description', 'Website', and 'Callback URL'. Each section has a text input field and a small explanatory note below it. The 'Name' field is filled with 'SentiAnalysis'. The 'Description' field is filled with 'Used for Sentiment Analysis of Tweets'. The 'Website' field is filled with 'https://greycampus.com'. The 'Callback URL' field is empty. The form is styled with a light grey background and white text.

Application Details

Name *

SentiAnalysis

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

Used for Sentiment Analysis of Tweets

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

https://greycampus.com

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens. (If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URL

Where should we return after successfully authenticating? OAuth 1.0a applications should explicitly specify their oauth_callback URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.

FIG 4.1.1 Create An Application

4.2 Registering App:

To register the twitter app, we need to create a new app <https://apps.twitter.com/>. On registering the app we will receive consumer key and consumer secret key. Next, From the configuration page of the app, we will get access token and access token secret, which will be used to get access to twitter on behalf of our application. We must keep these authentication tokens private as they can be misused. Best practice is to create a separate config file and keep these tokens.

The screenshot shows the Twitter Application Management page for an application named 'CodeLabsSentimentAnalysis'. The page has a blue header with the Twitter logo and 'Application Management'. Below the header, the application name is displayed in large bold text, with a 'Test OAuth' button to the right. A navigation bar contains links for 'Details', 'Settings', 'Keys and Access Tokens', and 'Permissions'. The 'Settings' tab is active, showing 'Application Settings'. A warning message states: 'Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.' The settings include: 'Consumer Key (API Key)' with a yellow box and a blue arrow pointing to it; 'Consumer Secret (API Secret)' with a yellow box and a yellow arrow pointing to it; 'Access Level' set to 'Read and write (modify app permissions)'; 'Owner' with a yellow box; and 'Owner ID' with a yellow box. At the bottom, the 'Application Actions' section contains two buttons: 'Regenerate Consumer Key and Secret' and 'Change App Permissions'.

FIG 4.2.1 Registering The Application

4.3 Accessing Data :

Twitter provides REST API's to connect with their service. We will use one python library to access the twitter REST API's called Tweepy. It provides wrapper methods to easily access twitter REST API. To install Tweepy we can use below command.:

Pip install tweepy

4.4 Storing Data:

Now we will access all tweet data from personal profile and store it for our analysis steps Tweepy library provides simple cursor interface to iterate through all the tweets and store them in file.

4.5 Preparing Data:-

Before we begin to analyze the twitter data, it's important to understand the structure of the tweet as well as pre-process the data to remove non-useful terms called stop-words. Preprocessing of data in data analysis is the very important step. Preprocessing is in the simple term means to take in the data and prepare the data for optimal output considering our requirement.

Tokenizing: the Tweet Tokenization is one of the most basic, yet most important, steps in text analysis. The purpose of tokenization is to split a stream of text into smaller units called tokens, usually words or phrases. We will use python NLTK library to tokenize the tweets. Even NLTK library needs some preprocessing steps to correctly tokenize @mentions and #hashtags. We use regular expressions to provide exceptions for mentions and hashtags.

Removing Stop-Words: Stop-word removal is one important step that should be considered during the pre-processing stages. Stop-words are most popular and common words of any language. While their use in the language is crucial, they don't usually convey a particular meaning, especially if taken out of context. This is the case of articles, conjunctions, some adverbs, etc. which are commonly called stop words. Some libraries provide default stop-words for different languages. NLTK library provides default stop-words for English language.

5.1 Output :



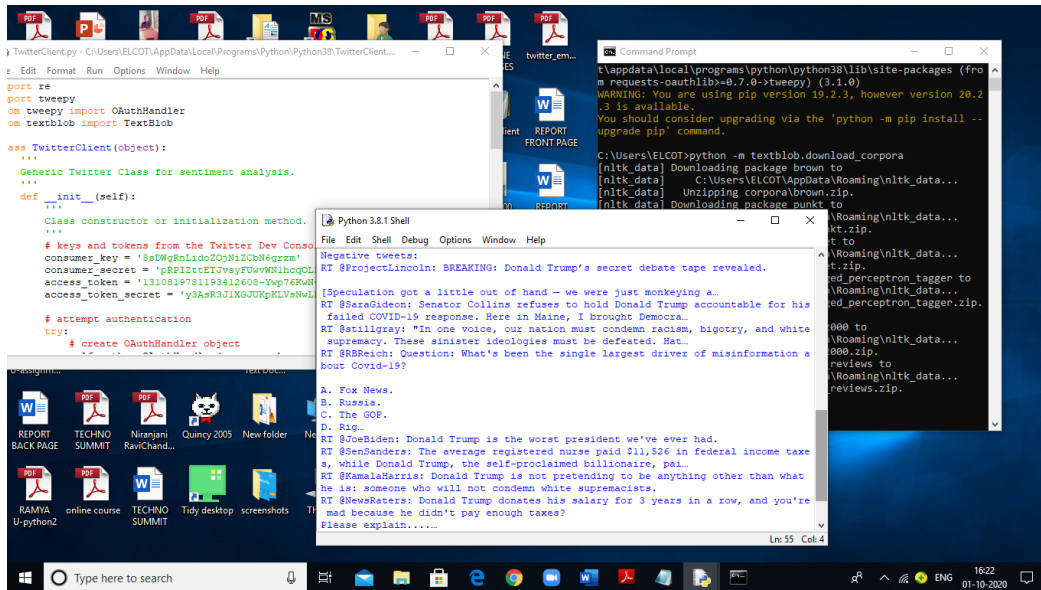


FIG 5.1.3 Negative Tweets

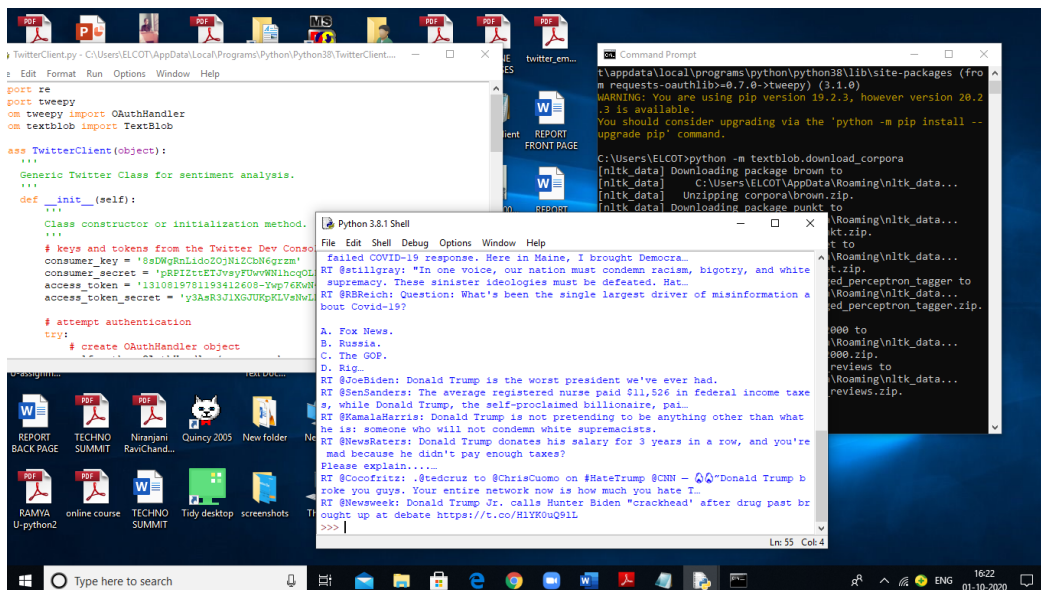


FIG 5.1.4 Tweets Explanation

CHAPTER 6

CONCLUSION

In this project, we started with very basics of Twitter data analysis. We explained for twitter app authentication using OAuth and Tweepy. Then we explained steps to collect historical data as well as streaming data. We then preprocessed the data using tokenizers. In the final step, we tried to execute a number of use cases to analyze the stored data. We represented results of analyzing most used terms for a data set, most used hashtags, most used mentions of user accounts on twitter and we also represented the bigrams i.e. two terms used frequently in our dataset. This project is introductory in nature and hence deals with basics of twitter data analysis using python. In future work, we will try to represent more advanced data analysis patterns decision making with more accurate results.

REFERENCES

1. J He, W Shen, P Divakaruni, L Wynter, R Lawrence, “Improving Traffic Prediction with Tweet Semantics”, Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pp. 1387–1393, August 3-9 2013.
2. A. Agarwal, B. Xie, I Vovsha, O. Rambow, R. Passonneau “Sentiment Analysis of Twitter Data” In the proceedings of Workshop on Language in Social Media, ACL, 2011
3. S Kumar, F Morstatter, H Liu, “Twitter Data Analytics” Springer Book 2013
4. A Mittal, A Goel, “Stock Prediction Using Twitter Sentiment Analysis”, Stanford University, 2011
5. “D Ediger, K Jiang, J Riedy, D. A. Bader “Massive Social Network Analysis: Mining Twitter for Social Good”, 39th International Conference on Parallel Processing 2010, pp. 583-593
6. <https://github.com/vivekwisdom/TwitterAnalysisApp>, Code repository of the sample application

APPENDIX

A. SOURCE 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 = '8sDWgRnLidoZOjNiZCbN6grzm'
        consumer_secret =
        'pRPIZttETJvsyFUwvWN1hcqOLkte7FNiDOt60YXGPofErx0dOI'
        access_token = '1310819781193412608-
        Ywp76KwN4ShAO4HDuvMqtE7w8YkyyI'
        access_token_secret =
        'y3AsR3J1XGJUKpKLVsNwLD6kCoBmN6hXXm7I2a6WCdHsV'
```

```

# 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+:\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 = 10):
    '''
    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(q = query, count = count)

# 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.TweepError as e:

        # print error (if any)
        print("Error : " + str(e))

def main():

    # creating object of TwitterClient Class
    api = TwitterClient()

    # calling function to get tweets
    tweets = api.get_tweets(query = 'Donald Trump', 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)))

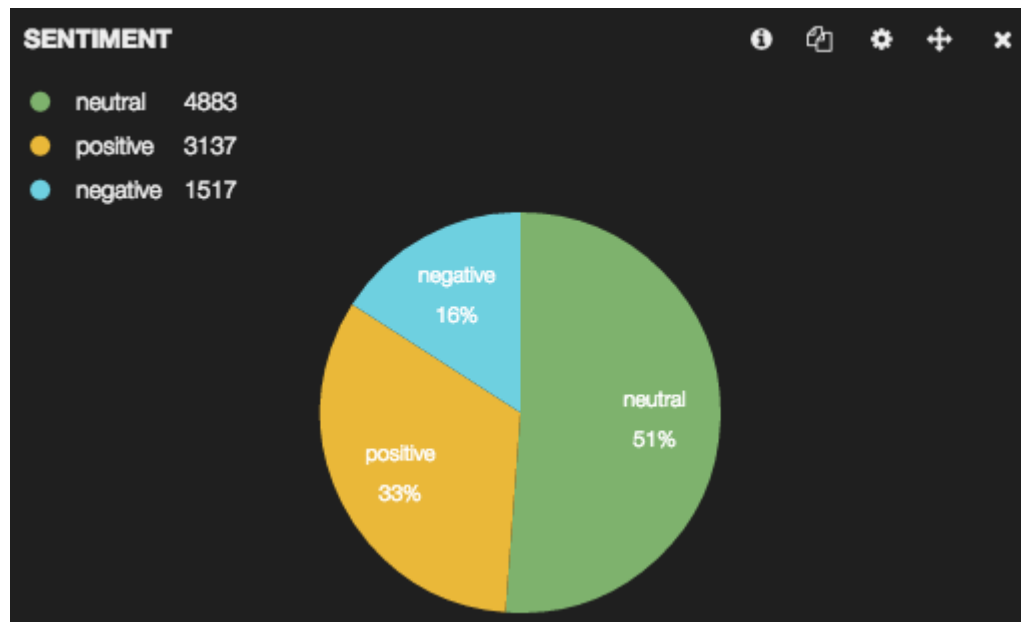
# printing first 5 positive tweets
print("\n\nPositive tweets:")
for tweet in ptweets[:10]:
    print(tweet['text'])

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

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

```

B.SCREEN SHOTS



Sentiment	Tweets
Negative	@united is the worst. Nonrefundable First class tickets? Oh because when you select Global/FC their system auto selects economy w/upgrade. @united I will not be flying you again
Neutral	@VirginAmerica my drivers license is expired by a little over a month. Can I fly Friday morning using my expired license? @VirginAmerica any plans to start flying direct from DAL to LAS?
Positive	@VirginAmerica done! Thank you for the quick response, apparently faster than sitting on hold ;) @united I appreciate your efforts getting me home!

