# Sentiment Analysis on tweets using machine learning

**Your Name here**

**Your Student ID**

**Semester**

Contents

[Sentiment Analysis on tweets using machine learning 1](#_Toc101801656)

[Introduction: 3](#_Toc101801657)

[Data: 3](#_Toc101801658)

[Objective: 3](#_Toc101801659)

[Significance: 4](#_Toc101801660)

[Methodology: 4](#_Toc101801661)

[Possible Approaches: 4](#_Toc101801662)

[Analysis of Results/Findings: 4](#_Toc101801663)

[Data cleaning: 8](#_Toc101801664)

[Sentiment analysis: 8](#_Toc101801665)

[References 14](#_Toc101801666)

# Introduction:

Sentiment analysis over Twitter offer associations a quick and powerful method for observing the publics' sentiments towards their brand, business, chiefs, and so on. A wide scope of elements and techniques for preparing opinion classifiers for Twitter datasets have been explored lately with fluctuating outcomes. We intend to perform sentiment analysis to investigate twitter data alluding to tweets connecting with gifts, raising money or good cause. This project covers methods and ways to deal with catch extremity of feelings of individuals towards giving for any purpose under exploratory data analysis. By utilizing Natural Language Processing Toolkit (NLTK) we decide if a tweet is of nonpartisan, positive or negative extremity. We are scrapping the data from twitter directly and will perform sentiment analysis to see if the tweet is positive or negative.

# Data:

We are collecting data directly from Twitter, we are using the Python’s module called Tweepy which is used for scrapping twitter tweets. Tweepy is an open-source Python bundle that gives you an exceptionally helpful method for getting to the Twitter API with Python. Tweepy incorporates a bunch of classes and strategies that address Twitter's models and API endpoints, and it straightforwardly handles different execution subtleties, for example, Data encoding and disentangling. We are using this module to fetch data of “Lockdown in UK”, we are initially scrapping only 2500 tweets related to our keyword. We will perform the sentimental analysis on these scrapped tweets and will find the positivity and negativity in the tweets regarding the Lockdown.

# Objective:

The rise of social media has given web clients a scene for communicating and sharing their considerations and assessments on a wide range of points and occasions. Twitter, with almost 600 million users and over 250 million messages for each day. Challenge is to separate significant data from twitter audits, since the idea of content is unstructured. Twitter need been the viewpoint of interest will a couple of experts on fundamental locales like forecast about evenhanded a couple of occasions, client brands, movie film industry, securities exchange, Notoriety from guaranteeing hotshots, etc. we are scrapping the data directly from Twitter using tweepy APIs by using a keyword ‘Lockdown’, we will perform sentiment analysis on the scrapped data to see the positivity and negativity in the tweets about lockdown.

# Significance:

Twitter sentiment examination permits you to monitor the thing's being said about your item or administration via web-based entertainment, and can assist you with distinguishing furious clients or pessimistic notices before they heighten. Opinion or sentiment examination apparatuses are fundamental to distinguish and figure out client sentiments. Organizations that utilization these devices to comprehend how clients feel can utilize it to work on CX. Feeling investigation instruments create bits of knowledge into how organizations can upgrade the client experience and further develop client care

# Methodology:

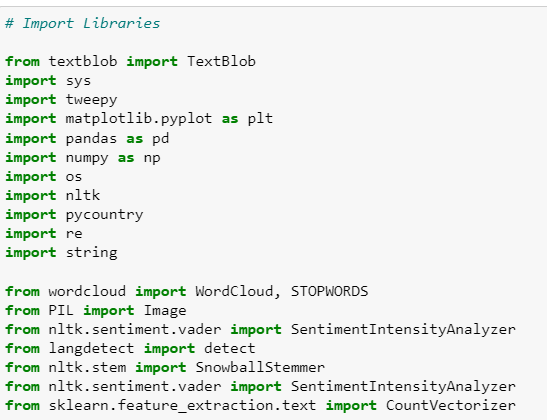
The overall methodology of this project is given below:

## Possible Approaches:

The goal of this project is to scrape Twitter tweets using the tweepy module, then apply NLP processing to identify that what people think about the lockdown? We are scrapping the tweets about lockdown in UK. We are going to use “Tweepy,” which is an easy-to-use Python library for accessing the Twitter API. The first thing that we needed is to have a Twitter developer account. With developer account, we cannot scrape the tweets from twitter. After scraping the tweets, we will apply the sentimental analysis to how people reacted to lockdown.

# Analysis of Results/Findings:

In this section, we will look at all the code we have performed and show the results we get. As the goal of this project is to scrape tweets from twitter by using tweepy and then do the sentimental analysis on the scraped tweets to see the positivity and negativity in the tweets, to achieve this we will first make a Python environment and then perform all the data analysis and data fetching.



We imported all the required libraries which are helpful in achieving what we want.

Tweepy upholds both OAuth 1a (application-user) and OAuth 2 (application-in particular) confirmation. Validation is dealt with by the tweepy.AuthHandler class.

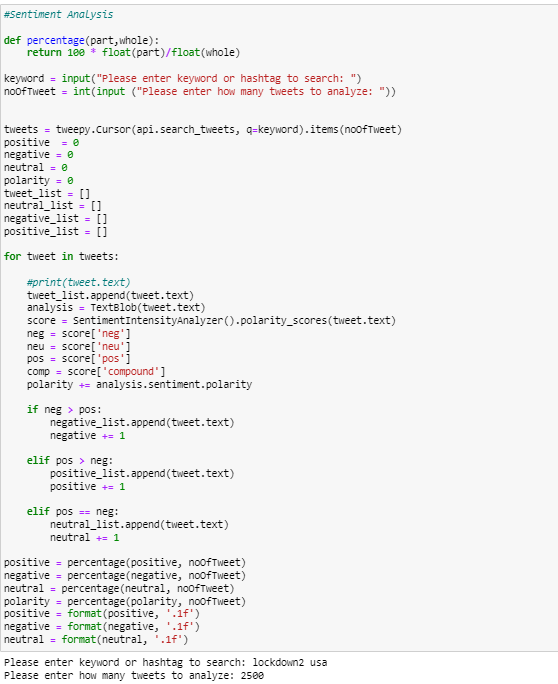
OAuth 2 is a technique for validation where an application makes API demands without the client setting. Utilize this strategy on the off chance that you simply need read-just admittance to public data. You first register the client application and gain a purchaser key and mystery. Then you make an AppAuthHandler case, passing in our purchaser key and mystery. For this, a twitter developer is compulsory to create. The authentication for twitter API is given below.



After the validation, we really want to utilize tweepy to get text and use Textblob to compute positive, negative, nonpartisan, extremity and compound boundaries from the text.

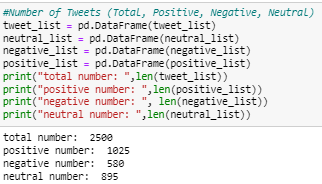
TextBlob is a Lexicon-based opinion analyzer It has some predefined rules or we can say word and weight word reference, where it has a scores that assistance to compute a sentence's extremity. That is the reason the Lexicon-based opinion analyzers are likewise called Rule-based feeling analyzers.

The next step is to get the tweets data from twitter by using a keyword or hashtags we want.

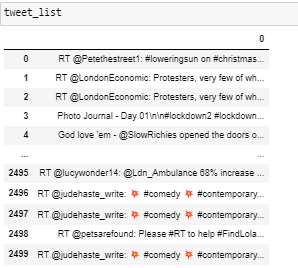


We are passing the keyword ‘lockdown UK’ and the number of tweets we want to scrape, in our case, we are scrapping 2500 tweets, the script will scrape all the 2500 tweets and the textblob will categorize the tweets as positive or negative and store the results in the separate lists.

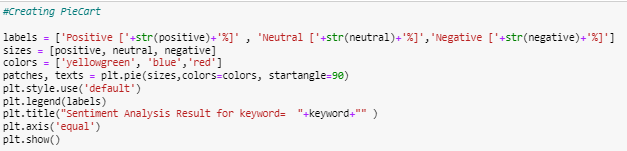
After scrapping the tweets and categorized them using the TextBlob, let’s see what we got.



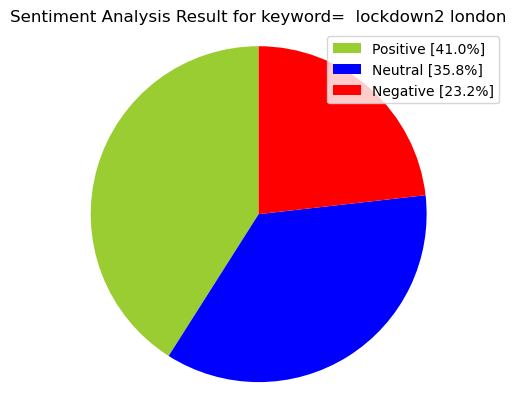
We got the total 2500 tweets in which 1025 are positive, 580 are negative and the rest are neutral.



Now let’s plot a pie chart to see the tweets visually.



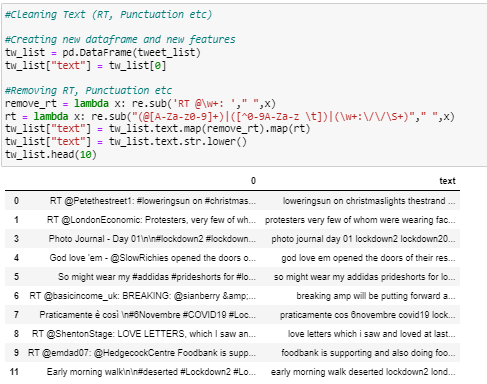
The above script will plot the following pie chart.



We can see the scrapped data more easily, the above chart shows the positivity, negativity and neutrality in percentage.

## Data cleaning:

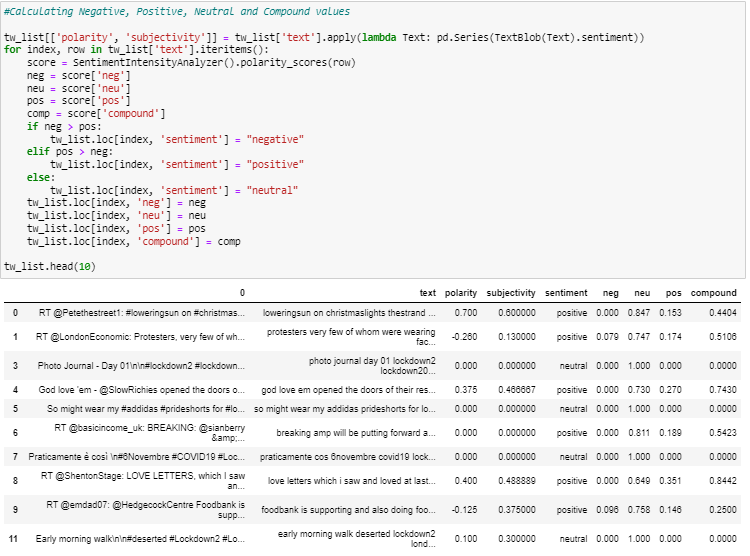
We have some duplicate tweets; these tweets need to be removed. We have a method called drop\_duplicates() which drop all the duplicate values in the data. After removing the duplicate tweets from the data, we got the total of 1281 tweets. We will make another data frame with some other features like ‘text’, etc.



We made another feature called text which have all the text data from the scrapped tweets without the irrelevant symbols and punctuations, etc.

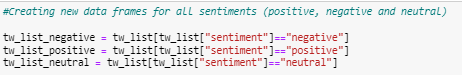
## Sentiment analysis:

We have cleaned the data by storing the tweets in another attribute which is called ‘text’ feature, we will use this feature to perform the sentiment analysis again to calculate polarity, subjectivity, sentiment, negative, positive, neutral and compound parameters. For all the mentioned parameters, we create new features to our data frame.

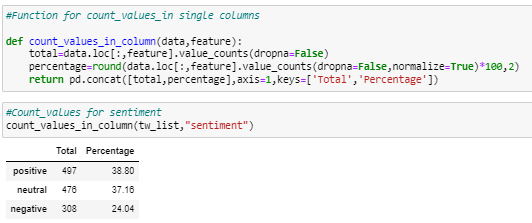


We are calculating the polarity and making some other features based on the parameters and storing the sentiments we get in the sentiment feature.

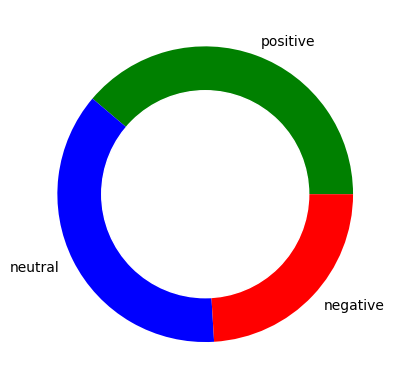
Now as we have got the sentiments, let’s split the into three different sentiments, for this purpose, we are separating the positive tweets, negative tweets and neutral tweets into three different data frames.



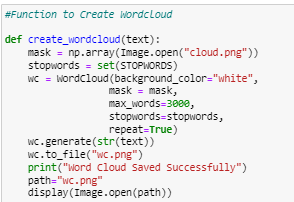
After separating the sentiments, lets count the values of sentiments and count the percentage of the sentiments.



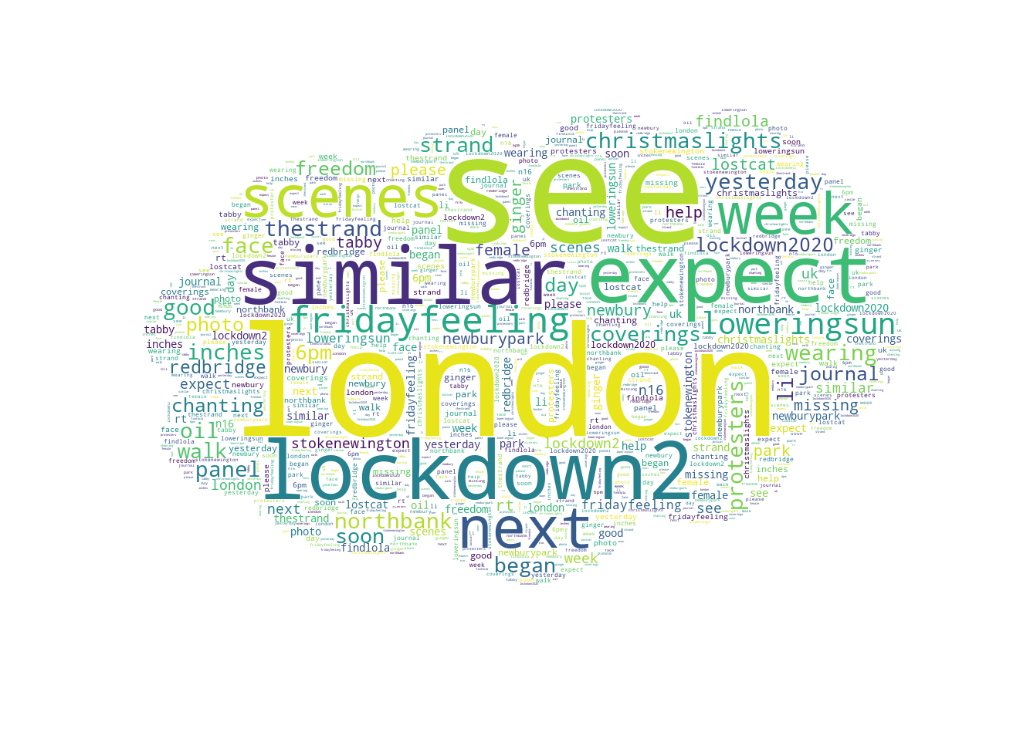
Let’s plot the pie chart again and see the sentiments visually.



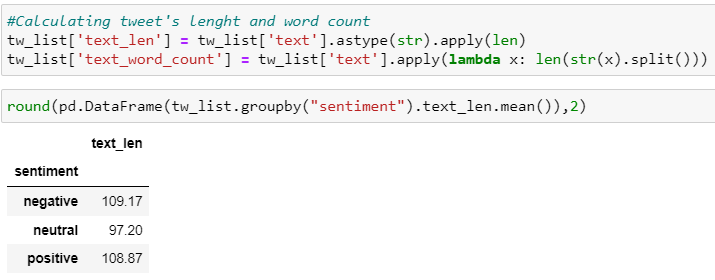
Presently you can plan to make worcloud utilizing 1281 tweets, So you can understand what words most utilized in these tweets. To make a worcloud, right off the bat we should characterize a capacity underneath, so you can utilize wordcloud again for all tweets, positive tweets, negative tweets and so forth.



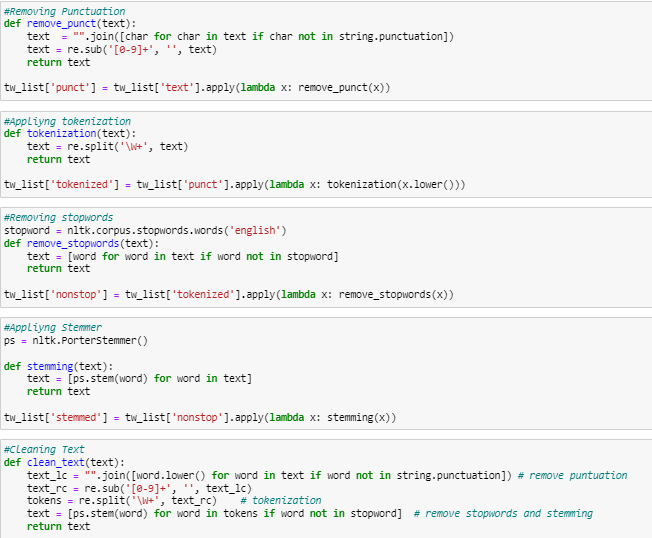
We will use the above function to plot the worcloud for all sentiments.



We calculated the tweet length and word count. The purpose of doing so is we can see the density of words and characters used in tweets based on different sentiment.



Applying count vectorizer gives the ability to preprocess your message information preceding creating the vector portrayal making it an exceptionally adaptable component portrayal module for message. After count vectorizer, it is feasible to examine the words with a few or anything you desire.



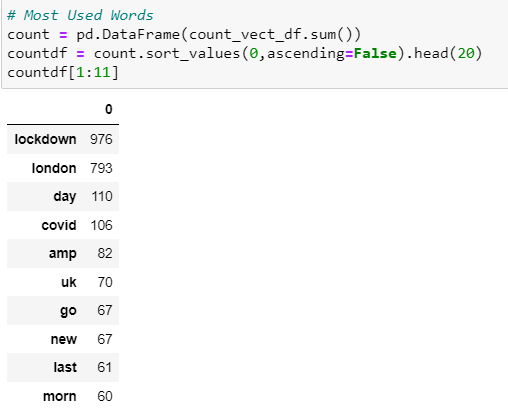
We are removing the punctuations in the first function, then tokenizing the data in the second function, removing the stop words in the third function and applying the stemming in the fourth function. Then we are applying all the functions in the fifth function. Bu doing so, we got the following data.



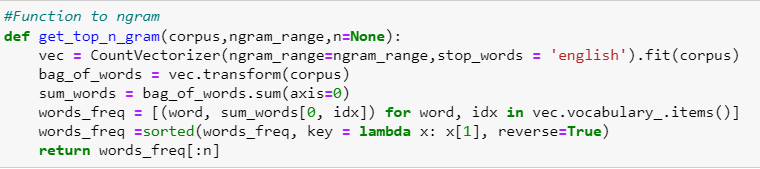
If you have a look at our data frame, you can see new features such as punct, tokenized, nonstop, stemmed. Now, we can apply coun vectorizer the see all 2966 unique words as a new feature.

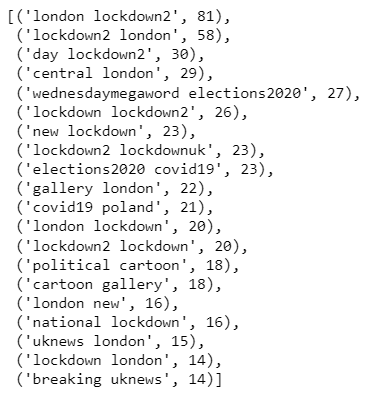


Let apply the ascending method to see the most used words.



Finally let’s build a n gram, structure n gram model helps us to predict most perhaps word that might follow this order. Firstly, let’s create a function then built n2\_bigram, n3\_trigram etc.





Finally, you can analyze sentiment using tweets and you can realize which words most used and which words used together.

# Conclusion:

In this project, we scraped the twitter data by using a keyword “Lockdown in London” by using the tweepy module. After scrapping the data, we performed the basic sentimental analysis on the tweets. In the research, we found out that 41% percent of the tweets were positive and 35% tweets were negative, the rest of the tweets were neutral. We performed some basis exploratory data analysis and tried to remove the duplicate tweets from the total number of 2500 tweets that we scrapped from twitter, after removing the duplicates, we are left with the total of 1281 tweets. We performed sentimental analysis again on the cleaned tweets and found out that there are the total of 38% tweets which are positive and 37% are negative, the rest are neutral. We draw some useful word clouds graphs and ngrams in the end of the project.

# References

Tweepy module: https://www.tweepy.org/