SIS Culminating Experience Project Report

on

Sentiment Analysis on Covid 19 Vaccines Booster Doses using Twitter API

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY

SPRING 2023

AT

UNC CHARLOTTE

by

Naga Mounika Gottumukkala

801276287

Table of Contents

1. Project Summary

"Sentiment Analysis on COVID-19 Vaccine's Booster doses using Twitter API."

The project's main goal is to analyze public opinion trends on COVID-19 booster shots using Tweets from Twitter, as well as to analyze people's positive, negative, and neutral attitudes based on tweets. The ability to identify booster shot sentiments from tweets would enable for improved decision-making in the current pandemic emergency.

During the spread of the COVID-19 Delta form, it was discovered that even after being fully vaccinated, people's immunity to the virus began to diminish. The Food and Drug Administration (FDA) developed COVID-19 vaccination booster shots, and the Centers for Disease Control and Prevention recommended them (CDC). These booster doses have been shown to reduce the risk of contracting the newly circulating variety.

The aim of the project is chiefly analyzing the trends in the public opinions on the COVID-19 booster shots based on Tweets from Twitter and analyzing the positive or negative sentiments of people based on tweets. These tweets would help analyze the pandemic situation and give informed decisions to handle the virus.

The results of the sentiment analysis will help to understand people's perception whether it's positive, negative, or neutral. This will further be helpful to explore the factors that motivate or discourage people to take the booster shot for the vaccine.

1.1 Motivation

The motivation behind choosing this project was to make people life's easier by analyzing the public opinion and sentiment about Covid 19 vaccines booster doses using data collected from Twitter. Social media has grown to be a popular platform for expressing our opinions nowadays, and there are many thoughts and feelings floating around concerning COVID booster dosages. The aim of the project is to expand our awareness of the worldwide pandemic and to engage as many individuals as we can by gaining insight into their opinions. For herd immunity to be achieved, this research is crucial. We benefit from the broad use of social media because it enables us to analyze and evaluate user postings to uncover aspects that might motivate others, particularly healthcare workers, to become immunized. It was an interesting project for all our teammates since we have not done something of this scale ever before.

1.2 Project Overview

"If you want to understand people, especially your customers, then you have to be able to possess a strong capability to analyze text." — Paul Hoffman. SARS-CoV-2, also known as COVID-19, has impacted over 96 million people worldwide, resulting in two million deaths. People have been panicked by a fast break out of COVID-19, and after a year of agony, disease, and destruction, the COVID-19 vaccinations have been pushed out as a gesture of relief.

There were arguments concerning the booster's efficacy, accessibility, and side effects all over the world, and grasping a subject and making good judgments has become tough because of this information overload. There is still a lot of confusion and disinformation about the vaccines' effectiveness and safety, which makes the COVID-19 booster doses seem unreliable.

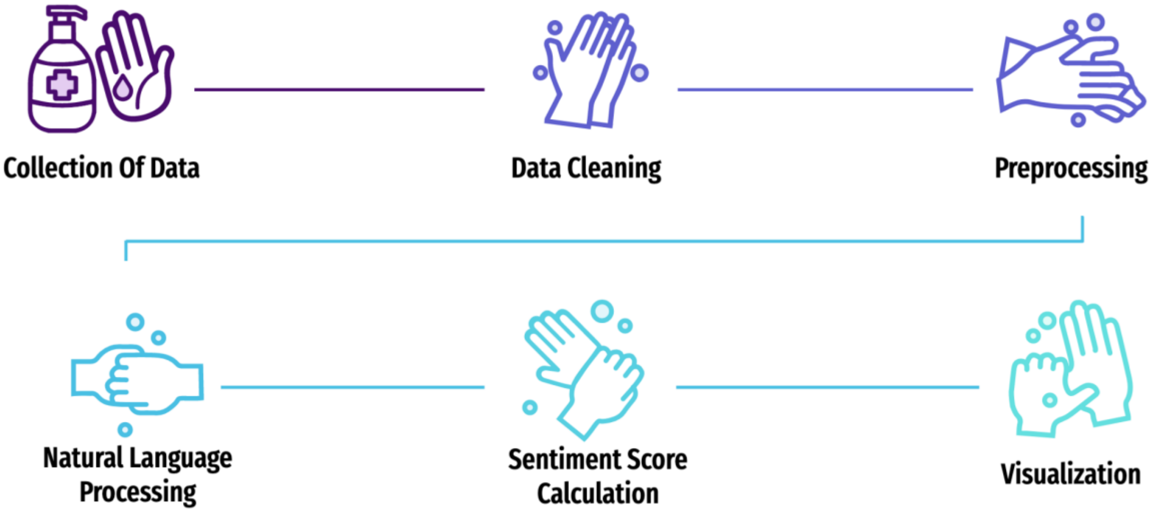
Today, social media has become a large forum for expressing our feelings, and there are a lot of emotions and sentiments about COVID booster doses out there. However, our exposure is restricted to the local news, thus the goal of this project is to broaden our view on the worldwide epidemic and to reach out to as many communities as possible by better understanding people's attitudes. This research is essential for achieving herd immunity.

Twitter has a massive following throughout the world, with 330 million daily active users. Twitter is used by people to communicate their ideas as well as convey information. The researchers have been able to identify user attitudes on nearly anything, including items, entertainment, politics, digital technology, and natural disasters, thanks to the quick sharing of user thoughts on Twitter.

We take advantage of social media's widespread use since it allows us to mine and analyze user postings to find elements that might motivate others, especially healthcare workers, to get vaccinated.

2. Project Approach

Process Flow



Collection of Data

Target data set is collected and created which is fetched from Twitter. Eventually, concentrating on the most appropriate data for discovery

Data Cleaning & Preprocessing

The data will be pre-processed to remove noise and simplify it. Handling of missing data-fields along with data transformation, tokenization, removing stop-words & punctuation.

Natural Language Processing Algorithms Application

Sentiment Analysis using ‘TextBLob’ along with ‘NLTK’ packages for text classification and categorization.

Sentiment Score Calculation & Visualization

Entities, topics, themes, and categories within a sentence or phrase receive weighted sentiment scores. Data visualization using bar charts, word clouds, pie charts along with region specific plotting of sentiment analysis on a map of the United States.

2.1 All About Data

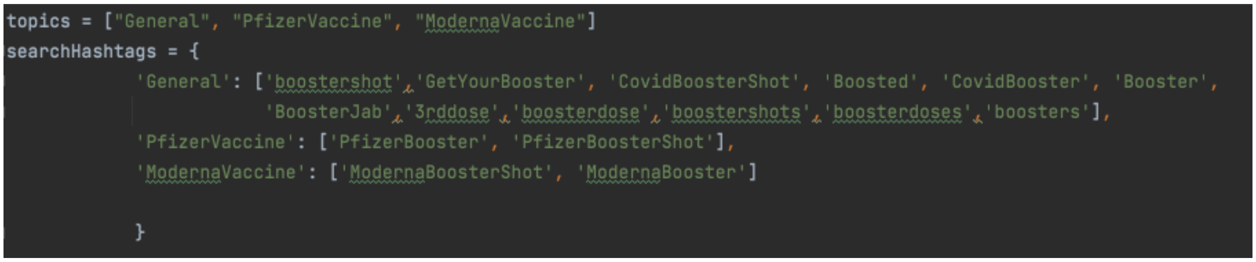
Extracting the Data

Twitter allows users to retrieve data from the last seven days. Our goal was to extract data to test our model and offer a prediction of good and negative tweets about COVID-19 Booster dosages. As a result, we'll require data from Twitter to feed into our model, and then provide the resulting data on socio-economic characteristics that have emerged since the vaccinations were made available to the public.

Extraction of high-quality data is one of the most critical and time-consuming activities in data mining and machine learning initiatives. Twitter provides a few APIs for accessing its data. The first step in data mining was to create a developer account and gain access to the Twitter API 1.1 version. We may establish a new project and obtain an API key, API key secret, access token, and access token secret after creating a Twitter developer account. We can use the Python Twitter module to establish secure connections to the Twitter API and access the different API endpoints supplied by Twitter using these credentials.

The API access credentials produced by the Twitter Developer account were sent to Twitter authentication handler instances. The Twitter search API is then used in conjunction with twitter cursor to scrape through the specified number of tweets and extract all relevant data.

Using the given code, we were able to retrieve over 30k tweets from Twitter using the Twitter API. Thus, extraction is often the procedure for extracting data from data sources for either storage or pre-processing. The extraction of high-quality data is critical. To preprocess and extract the data, we use a range of Python external packages. Tweets collected from Twitter are among the data in our study. In order to perform the authentication while authorizing the user, we have used the “oAuthentication_Login()” method. It follows the OAuth2.0 specification, which necessitates the use of a bearer token to verify the user's identity. We've developed a function called “fetchTwitterTweets()” to retrieve tweets from Twitter using Twitter API. This method downloads and saves tweets from the previous seven days to a MongoDB database as well as in a .csv file.



```
topics = ["General", "PfizerVaccine", "ModernaVaccine"]
searchHashtags = {
            'General': ['boostershot','GetYourBooster', 'CovidBoosterShot', 'Boosted', 'CovidBooster', 'Booster',
                        'BoosterJab','3rddose','boosterdose','boostershots','boosterdoses','boosters'],
            'PfizerVaccine': ['PfizerBooster', 'PfizerBoosterShot'],
            'ModernaVaccine': ['ModernaBoosterShot', 'ModernaBooster']

            }
```

Fig 1: Various Hashtags used in our dictionary along with major topics in a List

We've established a list of "hashtags" for which tweets must be retrieved. The following are the hashtags:

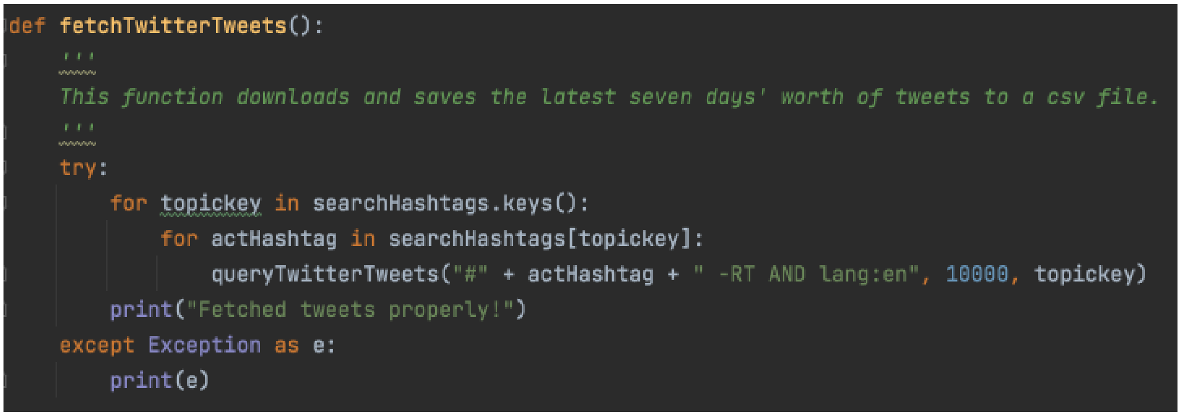
General Bucket:

'boostershot','GetYourBooster', 'CovidBoosterShot', 'Boosted',

'CovidBooster','Booster','BoosterJab','3rddose','boosterdose','boostershots','boosterdoses','boosters'. These are all general sets of hashtags which are used to fetch tweets as in majority the vaccine specific booster tweets are less as compared to these general set of tweets.

Pfizer Vaccine Bucket: 'PfizerBooster', 'PfizerBoosterShot'. These are all hashtags specific to the Pfizer booster vaccine as few tweets have been collected using this set of hashtags.

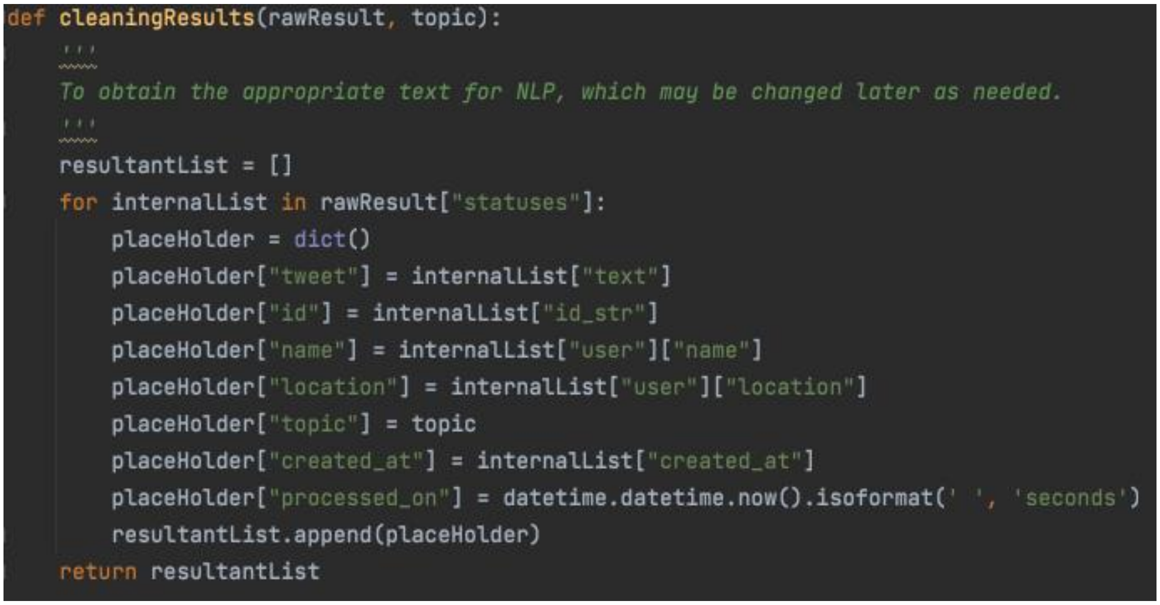
Moderna Vaccine Bucket: 'ModernaBoosterShot', 'ModernaBooster'. These are all hashtags specific to the Moderna booster vaccine as few tweets have been collected using this set of hashtags.



```python
def fetchTwitterTweets():
    '''
    This function downloads and saves the latest seven days' worth of tweets to a csv file.
    '''
    try:
        for topickey in searchHashtags.keys():
            for actHashtag in searchHashtags[topickey]:
                queryTwitterTweets("#" + actHashtag + " -RT AND lang:en", 10000, topickey)
        print("Fetched tweets properly!")
    except Exception as e:
        print(e)
```

Fig 2: Method which queries and stores the tweets of the past seven days

We have capped the value of tweets which are to be queried to a specific number 10,000 in the fetchTwitterTweets() method, of which all must be in English and all of those that don't include retweets. The fetchTwitterTweets() function takes all of the properties as arguments. Till the count limit of 10,000 is achieved, the fetchTwitterTweets() function queries and retrieves tweets for the supplied hashtags.

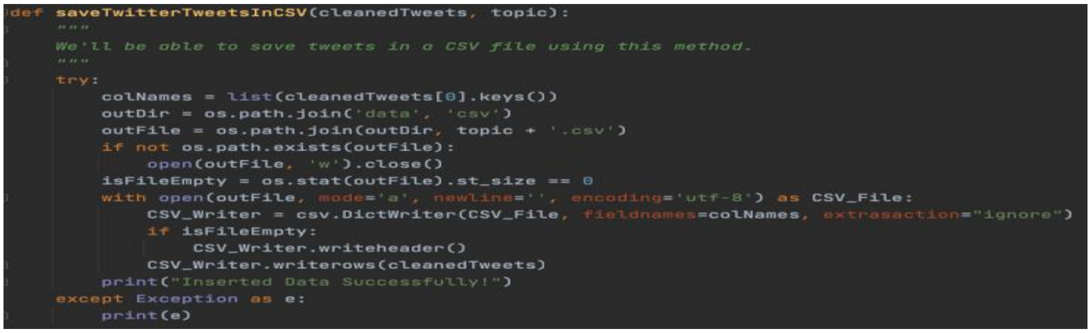


```python
def cleaningResults(rawResult, topic):
    '''
    To obtain the appropriate text for NLP, which may be changed later as needed.
    '''
    resultantList = []
    for internalList in rawResult["statuses"]:
        placeHolder = dict()
        placeHolder["tweet"] = internalList["text"]
        placeHolder["id"] = internalList["id_str"]
        placeHolder["name"] = internalList["user"]["name"]
        placeHolder["location"] = internalList["user"]["location"]
        placeHolder["topic"] = topic
        placeHolder["created_at"] = internalList["created_at"]
        placeHolder["processed_on"] = datetime.datetime.now().isoformat(' ', 'seconds')
        resultantList.append(placeHolder)
    return resultantList
```

Fig 3: A List which is manifested from the extracted data by selection technique.

We choose the appropriate fields from the tweets and save them in a.csv file in the form of row-column format where these fields are the columns as well as the database. Following are the selected columns as shown below:

1. The Tweet's wording in the form of text.
2. The tweet's identifier ID.
3. The identity/name of the person whose tweet has been fetched.
4. The location from where the tweet was tweeted originally.
5. The topic which means the bucket of our tweet.
6. The time when the tweet was created.
7. The time whenever the tweet was processed.



```
def saveTwitterTweetsInCSV(cleanedTweets, topic):
    """
    We'll be able to save tweets in a CSV file using this method.
    """
    try:
        colNames = list(cleanedTweets[0].keys())
        outDir = os.path.join('data', 'csv')
        outFile = os.path.join(outDir, topic + '.csv')
        if not os.path.exists(outFile):
            open(outFile, 'w').close()
        isFileEmpty = os.stat(outFile).st_size == 0
        with open(outFile, mode='a', newline='', encoding='utf-8') as CSV_File:
            CSV_Writer = csv.DictWriter(CSV_File, fieldnames=colNames, extrasaction="ignore")
            if isFileEmpty:
                CSV_Writer.writeheader()
            CSV_Writer.writerows(cleanedTweets)
        print("Inserted Data Successfully!")
    except Exception as e:
        print(e)
```

Fig 4: The snippet of code that saves the data to a .csv file.

The above-mentioned essential columns are obtained and saved in a .csv file. The gathered tweets are appended at the end of the file if it already exists. If our required .csv file does not already exist, a new one is created, and all our selected and filtered data is written in using a file writer.

2.2 Sentiment Analysis

Natural Language Processing (NLP) and Machine Learning are combined in Sentiment Analysis. It's used to figure out whether a data point is positive, negative, or neutral. It gives entities, themes, topics and categories within a sentence or phrase, weighted sentiment scores. Sentiment analysis aids businesses in gauging public opinion, tracking brand and product reputation, and gaining a better understanding of customer experience. Opinions are a great tool in determining the opinion or emotion of items or individuals because they are frequently stated on social media sites. These opinions have a significant impact on a product's or person's image. TextBlob, VADER, NLTK, CoreNLP, and others are examples of Natural Language Processing toolkits that can be used to construct meaningful sentiment analysis. "TextBlob" has been used in our project.

The following are the steps involved in fundamental sentiment analysis:

- We begin by dissecting each text document into its constituent elements. Sentences, phrases, tokens, and elements of speech are among these elements. The text in this case is a tweet.
- Then we look for any phrases or components that convey a sentiment.
- Each phrase is then given a sentiment score. The sentiment score is a number between -1 and 1
- We incorporate sentiment scores in a multi-layered form of sentiment analysis.

Sentiment Analysis has a wide range of real-world applications. Social media monitoring, customer support, customer feedback, brand monitoring and reputation management, market, and competitive research, and so on are examples of these uses. We will be able to discover variations in the overall perception of an opinion considering an individual or a product item using sentiment analysis.

Overview of Sentiment Polarity

The most important part of sentiment analysis is examining a body of text in order to decipher the opinions stated within it. We assign a positive, neutral or a negative value to this sentiment. The term "polarity" refers to this value.

Fig 5: Types of Polarities from Left-Right, Negative, Neutral & Positive

In sentiment analysis, polarity focuses on determining the sentiment direction. Simply put, it refers to the feelings portrayed in a sentence. It's a number between (-1) and (+1), with -1 indicating that it's a negative statement and +1 indicating a positive one. A statement with a polarity of zero is considered neutral. The sign (+ve/-ve) of the derived polarity score is frequently used to discern whether the overall broader emotion is positive, neutral, or negative.

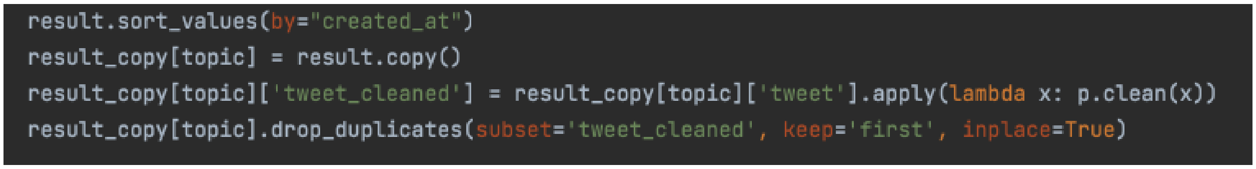
Building a Model for Sentiment Analysis

We'll need a labeled Twitter dataset to create the customized algorithm. A balanced dataset has an equal or nearly equal number of samples from both the positive and negative categories. Machine learning algorithms struggle with unbalanced training datasets because they are sensitive to the proportions of various classes. As a result, these algorithms prefer the class with the greatest number of observations (known as the majority class), which might lead to inaccurate results.

2.3 Data Preparation & Preprocessing

Data Cleaning Using Tweet Preprocessor

Data preprocessing is a crucial stage in the preparation of data for sentiment analysis. It has a direct impact on a project's success. If there are missing attributes, or missing attribute values, noise, or anomalies, and redundant or incorrect data, the data is considered to be unclean. Data in the real world is inconclusive, noisy, and irregular. If any of these are present, the quality of the results will deteriorate. Data preprocessing also minimizes the complexity of data that needs to be analyzed. We all know how clumsy tweet texts can be.



```
result.sort_values(by="created_at")
result_copy[topic] = result.copy()
result_copy[topic]['tweet_cleaned'] = result_copy[topic]['tweet'].apply(lambda x: p.clean(x))
result_copy[topic].drop_duplicates(subset='tweet_cleaned', keep='first', inplace=True)
```

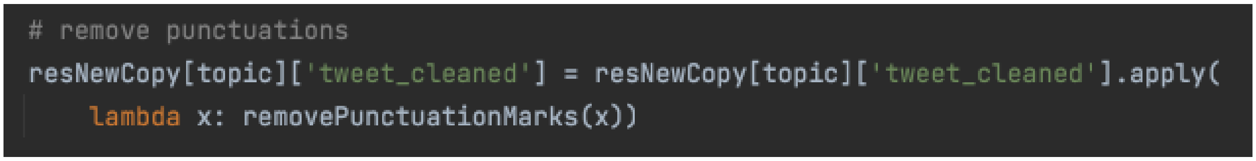
Fig 6: Using a Tweet Preprocessor to Clean Data

There is a Python library called "twitter-preprocessor" that allows us to conduct efficient tweet pre-processing. This command: "pip install tweet-processor" can be used to install it for the required dependencies. This package makes cleaning, tokenizing, and parsing tweets much simpler. Tweet-preprocessor eliminates URLs, hashtags, emojis, reserved words like RT and FAV, and mentions by default. Before performing sentiment analysis, we utilize this package to clean the data. We filter the data further after removing items like URLs, hashtags, emoticons, reserved words like RT and FAV, and mentions that don't affect the emotion of the text.

Because it is quite possible that a tweet will be retweeted by other people, the data acquired may contain duplicates. We remove duplicates from the acquired tweets to evade the same tweet. The method "drop duplicates ()" from the pandas' package is used. It aids in the removal of redundant values from the frame that we have created as a data frame. The parameters "subset", "keep" and "inplace" are used. Subset takes a column or a list of column labels in this case. Duplicates should be deleted from the columns in this table. The word "keep" is used to govern how values which are duplicates of each other are handled. It has three different values: "first," "last," and "false," with "first" being the default. We've used "first," which considers the first item to be unique and the remaining values to be duplicates. If "inplace" is set to True, it removes duplicate rows. Depending on the variables passed, the result of deleting duplicates is a dataframe with deleted duplicate rows. Other steps in the pre-processing process include:

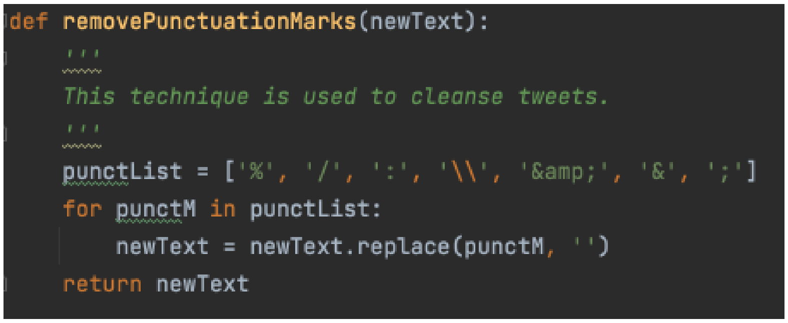
Removing Punctuation Marks

We delete punctuation marks such as ['%','/','':' ,'&', ';'] after data cleaning and deleting duplicate data and so forth. We defined the function "removePunctuationMarks()" to remove punctuation marks. This function removes punctuation and replaces it with a blank.



```
# remove punctuations
resNewCopy[topic]['tweet_cleaned'] = resNewCopy[topic]['tweet_cleaned'].apply(
    lambda x: removePunctuationMarks(x))
```

Fig 7: Calling the removePunctuationMarks() method.

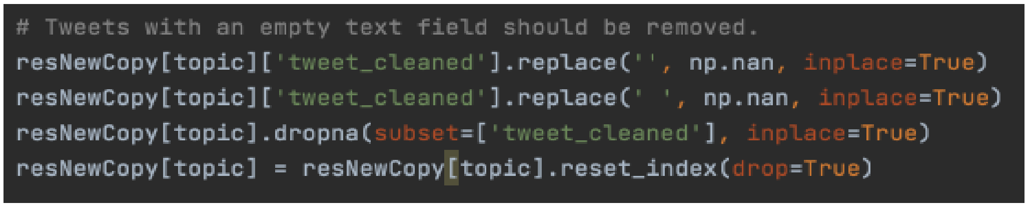


```
def removePunctuationMarks(newText):
    '''
    This technique is used to cleanse tweets.
    '''
    punctList = ['%', '/', ':', '\\', '&amp;', '&', ';']
    for punctM in punctList:
        newText = newText.replace(punctM, '')
    return newText
```

Fig 8: The actual method logic for removing Punctuation marks

Removal of Missing Data

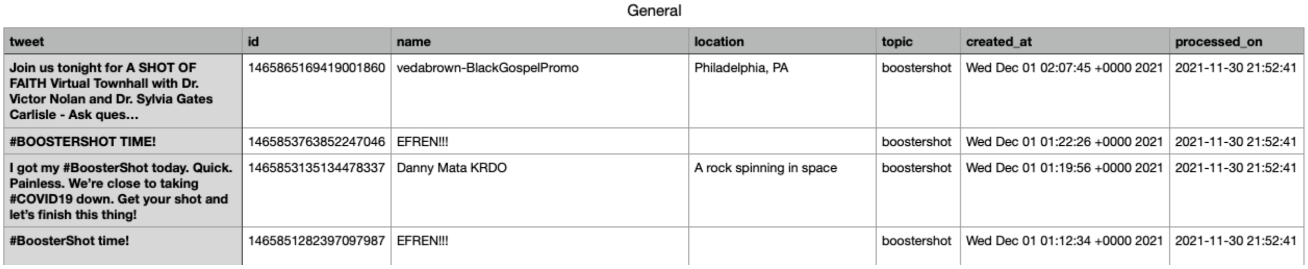
Rows containing missing data must also be removed. We remove rows that have tweets with empty text fields in this section. For this, we utilize the pandas function dropna(). Subset and inplace are the two parameters that we pass as parameters to this function. Subset takes the labels of a column or a list of columns. Because "inplace" is set to True, the operation is performed in place. The existing index of the dataframe is then removed and replaced with an index of ascending integers. The reset index () method from the pandas package is used to accomplish this.



```
# Tweets with an empty text field should be removed.
resNewCopy[topic]['tweet_cleaned'].replace('', np.nan, inplace=True)
resNewCopy[topic]['tweet_cleaned'].replace(' ', np.nan, inplace=True)
resNewCopy[topic].dropna(subset=['tweet_cleaned'], inplace=True)
resNewCopy[topic] = resNewCopy[topic].reset_index(drop=True)
```

Fig 9: Using a line of code, you can drop tweets with an empty text box.

After the data has been pre-processed, it looks like this:



General

tweet	id	name	location	topic	created_at	processed_on
Join us tonight for A SHOT OF FAITH Virtual Townhall with Dr. Victor Nolan and Dr. Sylvia Gates Carlisle - Ask ques…	1465865169419001860	vedabrown-BlackGospelPromo	Philadelphia, PA	boostershot	Wed Dec 01 02:07:45 +0000 2021	2021-11-30 21:52:41
#BOOSTERSHOT TIME!	1465853763852247046	EFREN!!!		boostershot	Wed Dec 01 01:22:26 +0000 2021	2021-11-30 21:52:41
I got my #BoosterShot today. Quick. Painless. We're close to taking #COVID19 down. Get your shot and let's finish this thing!	1465853135134478337	Danny Mata KRDO	A rock spinning in space	boostershot	Wed Dec 01 01:19:56 +0000 2021	2021-11-30 21:52:41
#BoosterShot time!	1465851282397097987	EFREN!!!		boostershot	Wed Dec 01 01:12:34 +0000 2021	2021-11-30 21:52:41

Removing the stop words

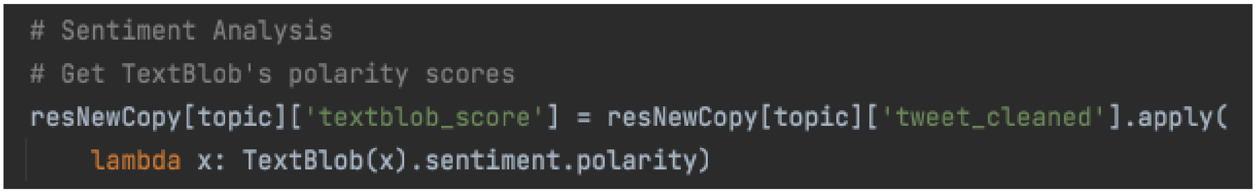
Stop words are the most frequently used terms in a text. Words like "the," "is," and "and" are among them. The most common stop words are included in the NLTK corpus by default. Stop words are used in NLP and text mining applications to remove unnecessary terms, enabling programs to focus on the key words instead.

TextBlob

TextBlob is an inbuilt python library that is primarily used for interpreting and analyzing textual data. It provides an API kind of service to perform small but important Natural Language Processing tasks such as parts of speech tagging (POS) and noun phrase extraction. These small tasks are then useful in performing sentiment analysis. Natural Language ToolKit (NLTK) is used extensively by TextBlob to complete its objectives. NLTK is indeed a library that allows users to work with categorization, classification, and a variety of other tasks by providing easy access to many lexical resources. TextBlob is a basic package that allows for extensive textual data analysis and operations. TextBlob is an excellent package for folks who have just had a hand on with Natural Language Processing and sentiment analysis.

2.4 Analyzing the Sentiment

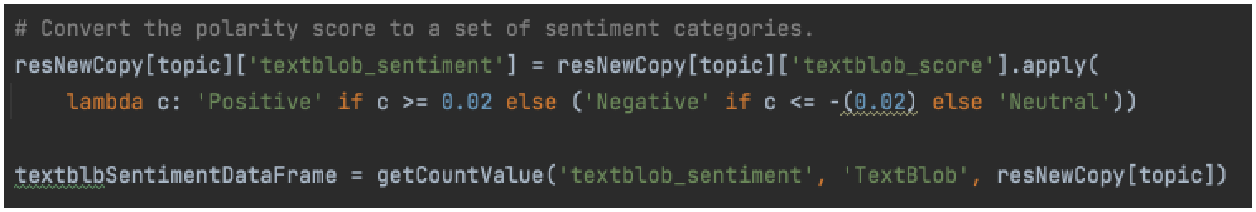
We submit data to the sentiment analyzer after it has been pre-processed. We used TextBlob as a sentiment analysis tool in this case. For each tweet, TextBlob returns a value termed as the polarity score. The polarity score is stored in the "textblob_score" column in this project. The polarity score is indeed a floating-point number between -1 and +1, with -1 denoting the most negative polarity and +1 denoting the most positive polarity.



```
# Sentiment Analysis
# Get TextBlob's polarity scores
resNewCopy[topic]['textblob_score'] = resNewCopy[topic]['tweet_cleaned'].apply(
    lambda x: TextBlob(x).sentiment.polarity)
```

Fig 10: TextBlob is used to clean up the data.

TextBlob provides a property called sentiment.polarity that can be used to calculate the polarity score. We translate the polarity score into sentiment categories after collecting the sentiment score.

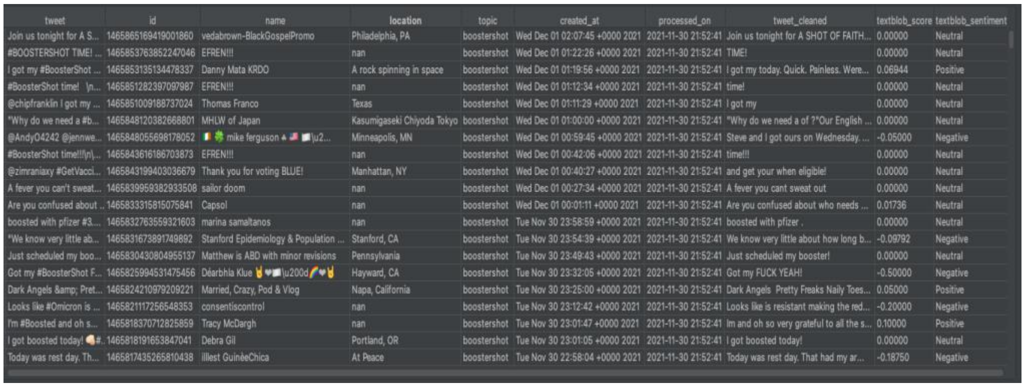


```
# Convert the polarity score to a set of sentiment categories.
resNewCopy[topic]['textblob_sentiment'] = resNewCopy[topic]['textblob_score'].apply(
    lambda c: 'Positive' if c >= 0.02 else ('Negative' if c <= -(0.02) else 'Neutral'))

textblbSentimentDataFrame = getCountValue('textblob_sentiment', 'TextBlob', resNewCopy[topic])
```

Fig 11: Sentiment Categorization Thresholds and Values

The tweet is classed as having a positive sentiment if the computed sentiment polarity is greater than or equal to 0.02. The tweet is classed as having a negative sentiment if the obtained sentiment polarity is less than or equal to (-0.02). Aside from that, the tweets polarity between (-0.02 to 0.02) exclusive of borders is deemed neutral.



tweet	id	name	location	topic	created_at	processed_on	tweet_cleaned	textblob_score	textblob_sentiment
Join us tonight for A S...	1465865169419001860	vedabrown-BlackGospelPromo	Philadelphia, PA	boostershot	Wed Dec 01 02:07:45 +0000 2021	2021-11-30 21:52:41	Join us tonight for A SHOT OF FAITH...	0.00000	Neutral
#BOOSTERSHOT TIME! ...	1465853763852247046	EFREN!!!	nan	boostershot	Wed Dec 01 01:22:26 +0000 2021	2021-11-30 21:52:41	TIME!	0.00000	Neutral
I got my #BoosterShot ...	1465853135134478337	Danny Mata KRDO	A rock spinning in space	boostershot	Wed Dec 01 01:19:56 +0000 2021	2021-11-30 21:52:41	I got my today. Quick. Painless. Were...	0.06944	Positive
#BoosterShot time! \n...	1465851282397097987	EFREN!!!	nan	boostershot	Wed Dec 01 01:12:34 +0000 2021	2021-11-30 21:52:41	time!	0.00000	Neutral
@chipfranklin I got my ...	1465851009188737024	Thomas Franco	Texas	boostershot	Wed Dec 01 01:11:29 +0000 2021	2021-11-30 21:52:41	I got my	0.00000	Neutral
"Why do we need a #b...	1465848120382668801	MHLW of Japan	Kasumigaseki Chiyoda Tokyo	boostershot	Wed Dec 01 01:00:00 +0000 2021	2021-11-30 21:52:41	"Why do we need a of ?"Our English ...	0.00000	Neutral
@AndyO4242 @jennwe...	1465848055698178052	🇮🇪 🍀 mike ferguson ♣ 🇺🇸 🏳\u2...	Minneapolis, MN	boostershot	Wed Dec 01 00:59:45 +0000 2021	2021-11-30 21:52:41	Steve and I got ours on Wednesday. ...	-0.05000	Negative
#BoosterShot time!!!\n\...	1465843616186703873	EFREN!!!	nan	boostershot	Wed Dec 01 00:42:06 +0000 2021	2021-11-30 21:52:41	time!!!	0.00000	Neutral
@zimraniaxy #GetVacci...	1465843199403036679	Thank you for voting BLUE!	Manhattan, NY	boostershot	Wed Dec 01 00:40:27 +0000 2021	2021-11-30 21:52:41	and get your when eligible!	0.00000	Neutral
A fever you can't sweat...	1465839959382933508	sailor doom	nan	boostershot	Wed Dec 01 00:27:34 +0000 2021	2021-11-30 21:52:41	A fever you cant sweat out	0.00000	Neutral
Are you confused about ...	1465833315815075841	Capsol	nan	boostershot	Wed Dec 01 00:01:11 +0000 2021	2021-11-30 21:52:41	Are you confused about who needs ...	0.01736	Neutral
boosted with pfizer #3....	1465832763559321603	marina samaltanos	nan	boostershot	Tue Nov 30 23:58:59 +0000 2021	2021-11-30 21:52:41	boosted with pfizer .	0.00000	Neutral
"We know very little ab...	1465831673891749892	Stanford Epidemiology & Population ...	Stanford, CA	boostershot	Tue Nov 30 23:54:39 +0000 2021	2021-11-30 21:52:41	We know very little about how long b...	-0.09792	Negative
Just scheduled my boo...	1465830430804955137	Matthew is ABD with minor revisions	Pennsylvania	boostershot	Tue Nov 30 23:49:43 +0000 2021	2021-11-30 21:52:41	Just scheduled my booster!	0.00000	Neutral
Got my #BoosterShot F...	1465825994531475456	Déarbhla Klue 🤘❤🏳\u200d🌈❤🤘	Hayward, CA	boostershot	Tue Nov 30 23:32:05 +0000 2021	2021-11-30 21:52:41	Got my FUCK YEAH!	-0.50000	Negative
Dark Angels &amp; Pret...	1465824210979209221	Married, Crazy, Pod & Vlog	Napa, California	boostershot	Tue Nov 30 23:25:00 +0000 2021	2021-11-30 21:52:41	Dark Angels Pretty Freaks Naily Toes...	0.05000	Positive
Looks like #Omicron is ...	1465821117256548353	consentiscontrol	nan	boostershot	Tue Nov 30 23:12:42 +0000 2021	2021-11-30 21:52:41	Looks like is resistant making the red...	-0.20000	Negative
I'm #Boosted and oh s...	1465818370712825859	Tracy McDargh	nan	boostershot	Tue Nov 30 23:01:47 +0000 2021	2021-11-30 21:52:41	Im and oh so very grateful to all the s...	0.10000	Positive
I got boosted today! 👊#..	1465818191653847041	Debra Gil	Portland, OR	boostershot	Tue Nov 30 23:01:05 +0000 2021	2021-11-30 21:52:41	I got boosted today!	0.00000	Neutral
Today was rest day. Th...	1465817435265810438	illest GuinéeChica	At Peace	boostershot	Tue Nov 30 22:58:04 +0000 2021	2021-11-30 21:52:41	Today was rest day. That had my ar...	-0.18750	Negative

Fig 12: Data after acquiring the polarity and category of its sentiment.

TextBlob outputs can be categorized in two ways. Firstly, polarity where the output is in the range of -1.0 to 1.0. -1.0 suggests a negative polarity and 1.0 suggests a positive polarity. Secondly, subjectivity where the output is in the range of 0.0 to 1.0. 0.0 would suggest objectivity and 1.0 would suggest subjectivity.

2.5 Results and Outcomes

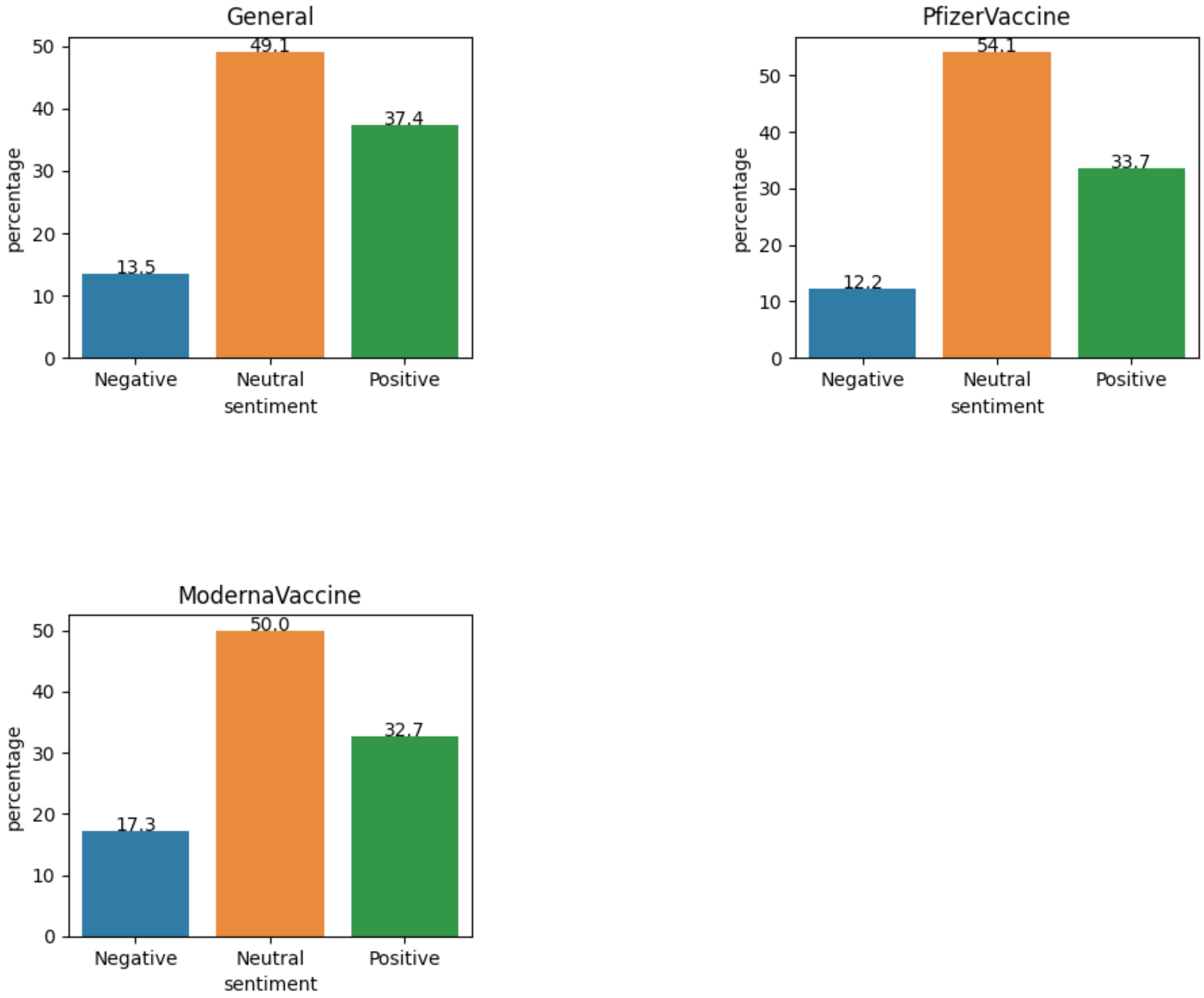
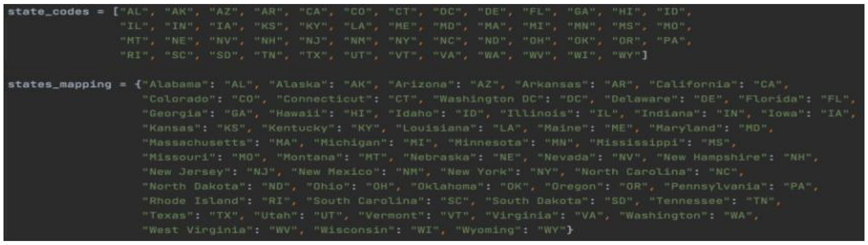


Fig 13: Bar Graph showing positive and negative sentiment for General, Pfizer and Moderna vaccine

The resultant graph above shows the output from the model we have created after the input has been fed to it. The graph describes the positive and negative tweet distribution. As seen the graph has been dominated with a positive response with very less negativity spread around. On observing we can also see that the Moderna vaccine booster dose has received a more negative sentiment as compared to the Pfizer booster dose.

Booster vaccines are seen differently by people from different states in the United States. During this analysis, we also discover people's attitudes toward booster vaccine doses by state. If the user is disclosing his or her location, we can get the location from which the tweets were posted when we extracted the tweets from Twitter.

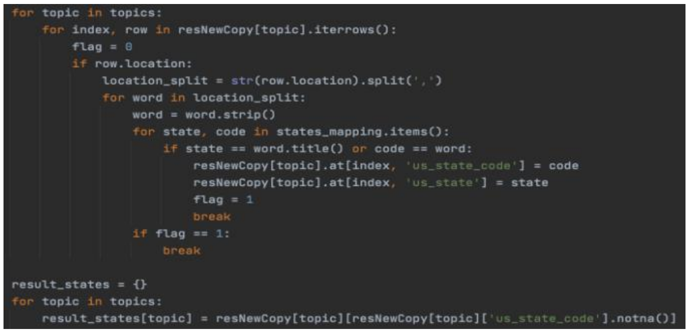
This data can be blank at times. It may also consist of just the nation name, just the city name, or just the city and state names separated by a comma at other times. We construct a list of state codes as well as a dictionary that holds the state code to state name mapping. This is what we utilize to extract tweets that have a value in their location records. These are derived from the data that has been cleansed and processed. The state codes are shown in the diagram below, along with a dictionary containing the state code as a key and the state name as a value.



```
state_codes = ["AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DC", "DE", "FL", "GA", "HI", "ID",
               "IL", "IN", "IA", "KS", "KY", "LA", "ME", "MD", "MA", "MI", "MN", "MS", "MO",
               "MT", "NE", "NV", "NH", "NJ", "NM", "NY", "NC", "ND", "OH", "OK", "OR", "PA",
               "RI", "SC", "SD", "TN", "TX", "UT", "VT", "VA", "WA", "WV", "WI", "WY"]

states_mapping = {"Alabama": "AL", "Alaska": "AK", "Arizona": "AZ", "Arkansas": "AR", "California": "CA",
                  "Colorado": "CO", "Connecticut": "CT", "Washington DC": "DC", "Delaware": "DE", "Florida": "FL",
                  "Georgia": "GA", "Hawaii": "HI", "Idaho": "ID", "Illinois": "IL", "Indiana": "IN", "Iowa": "IA",
                  "Kansas": "KS", "Kentucky": "KY", "Louisiana": "LA", "Maine": "ME", "Maryland": "MD",
                  "Massachusetts": "MA", "Michigan": "MI", "Minnesota": "MN", "Mississippi": "MS",
                  "Missouri": "MO", "Montana": "MT", "Nebraska": "NE", "Nevada": "NV", "New Hampshire": "NH",
                  "New Jersey": "NJ", "New Mexico": "NM", "New York": "NY", "North Carolina": "NC",
                  "North Dakota": "ND", "Ohio": "OH", "Oklahoma": "OK", "Oregon": "OR", "Pennsylvania": "PA",
                  "Rhode Island": "RI", "South Carolina": "SC", "South Dakota": "SD", "Tennessee": "TN",
                  "Texas": "TX", "Utah": "UT", "Vermont": "VT", "Virginia": "VA", "Washington": "WA",
                  "West Virginia": "WV", "Wisconsin": "WI", "Wyoming": "WY"}
```

Fig 14: A dictionary that maps state codes to state names, as well as a list of state codes.



```
for topic in topics:
    for index, row in resNewCopy[topic].iterrows():
        flag = 0
        if row.location:
            location_split = str(row.location).split(',')
            for word in location_split:
                word = word.strip()
                for state, code in states_mapping.items():
                    if state == word.title() or code == word:
                        resNewCopy[topic].at[index, 'us_state_code'] = code
                        resNewCopy[topic].at[index, 'us_state'] = state
                        flag = 1
                        break
                if flag == 1:
                    break

result_states = {}
for topic in topics:
    result_states[topic] = resNewCopy[topic][resNewCopy[topic]['us_state_code'].notna()]
```

Fig 15: Code for picking tweets that have complete location information.

The code snippet above illustrates the extraction of tweets based on the availability of location records by utilizing the dictionary "states mapping." We exhibit graphs demonstrating the semantic orientation towards vaccines after getting the tweets with a value in the location field.

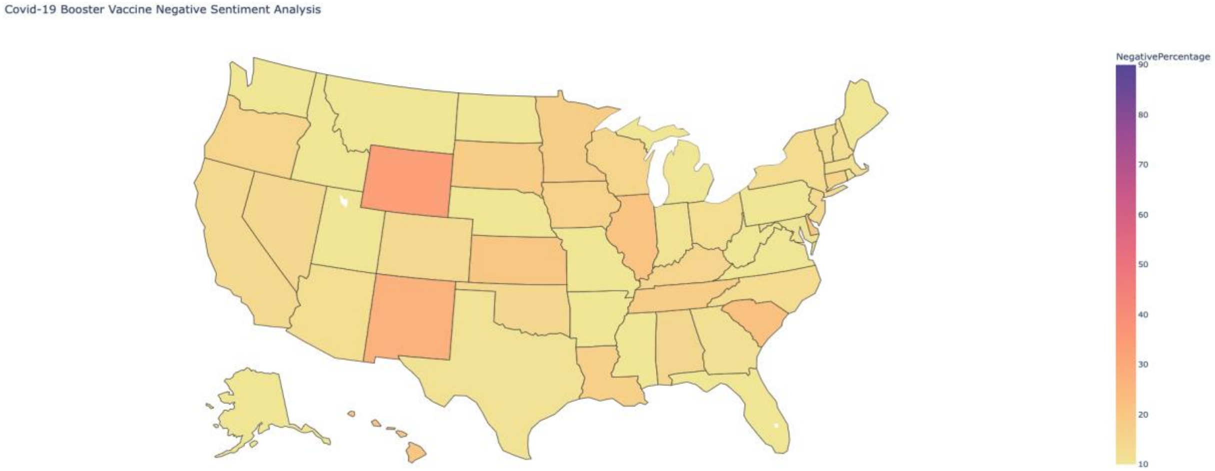
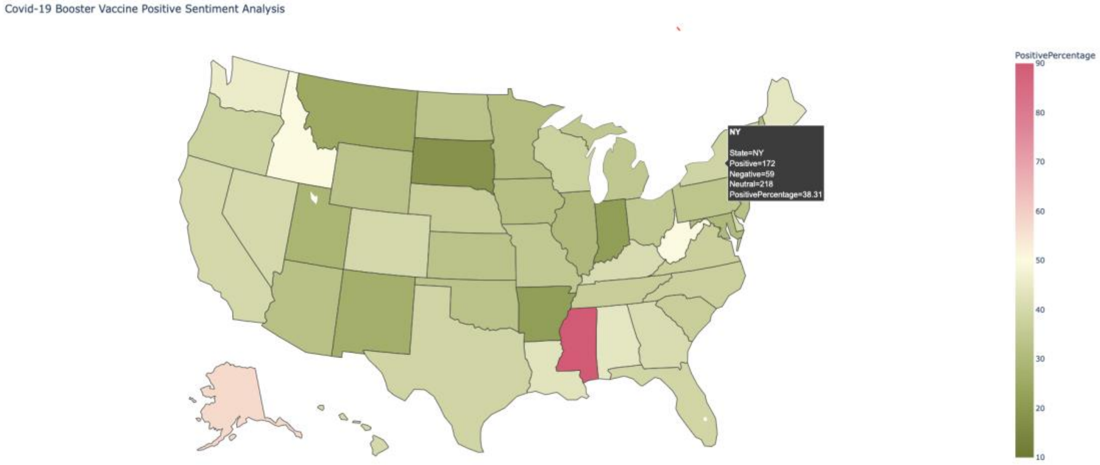


Fig 16: Heatmap of the percentage of negative sentiment in United States of America

The above figure depicts the percentage of negative sentiment regarding the booster shot of the general vaccine. The plot clearly depicts that the state Wyoming has a negative sentiment of around 70% which is the most among all the 50 states of United States of America.States like Nevada, New Mexico,Kansas,South Dakota and South Carolina have a negative sentiment of around 40% which is comparatively less compared to Wyoming. Other states have negative sentiment of 10-20% approximately.



NY

State=NY
Positive=172
Negative=59
Neutral=218
PositivePercentage=38.31

Fig 17: Heatmap of the percentage of positive sentiment in United States of America

The above figure depicts the percentage of positive sentiment regarding the booster shot of the general vaccine. The plot clearly depicts that the state Mississippi has a positive sentiment of around 90% which is the most among all the 50 states of United States of America.As depicted in the heatmap, the state of NY has positive sentiment percentage of 33.18%

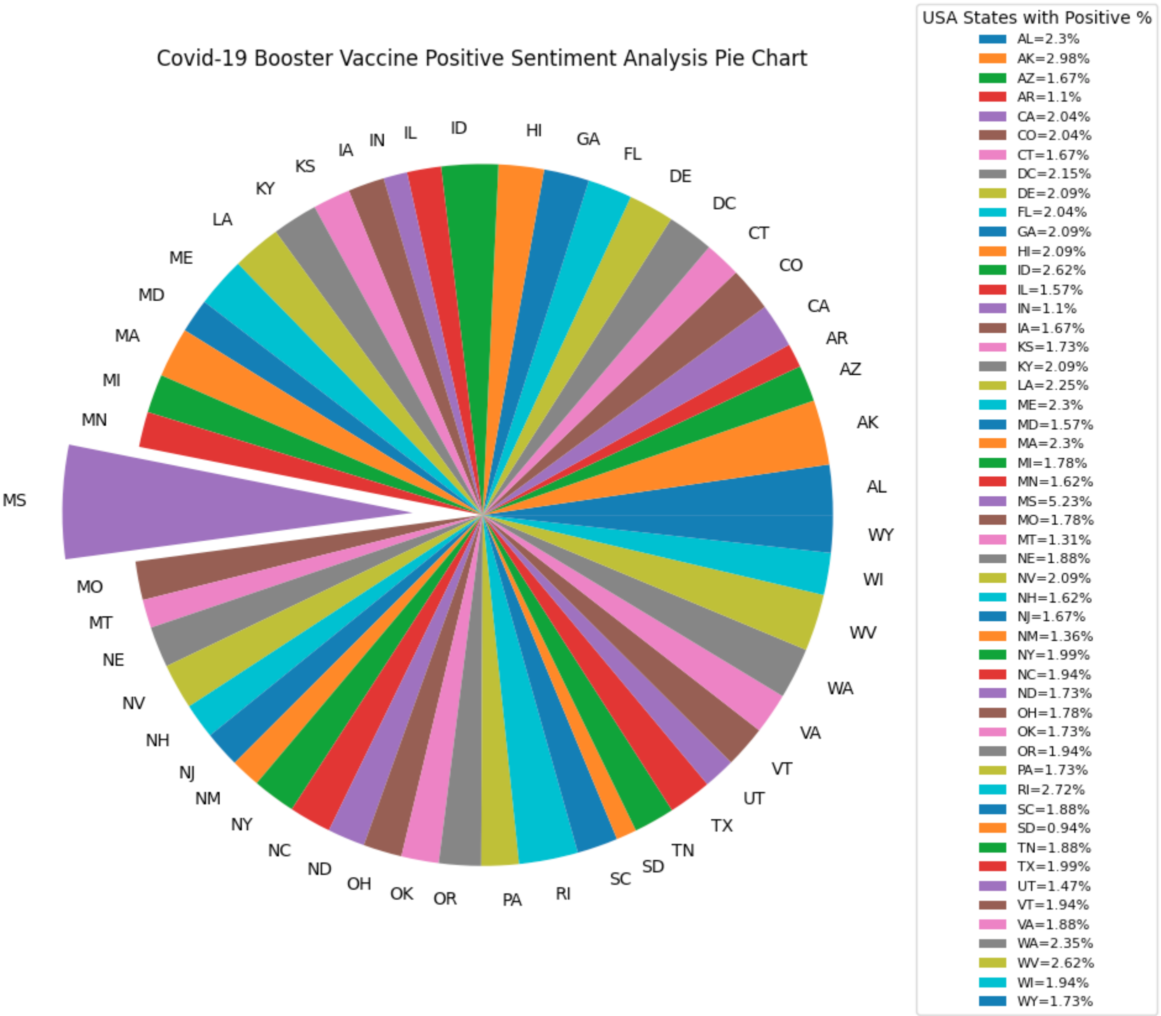


Fig 18: Pie Chart depicting positive sentiment percentage per State

From the above pie chart, we can see that the positive sentiment rate for Mississippi is the highest with 5.23% among all the 50 states in the United States of America. This implies that there is a lot of positivity about Booster Vaccine shot as analyzed by the tweet sentiment.

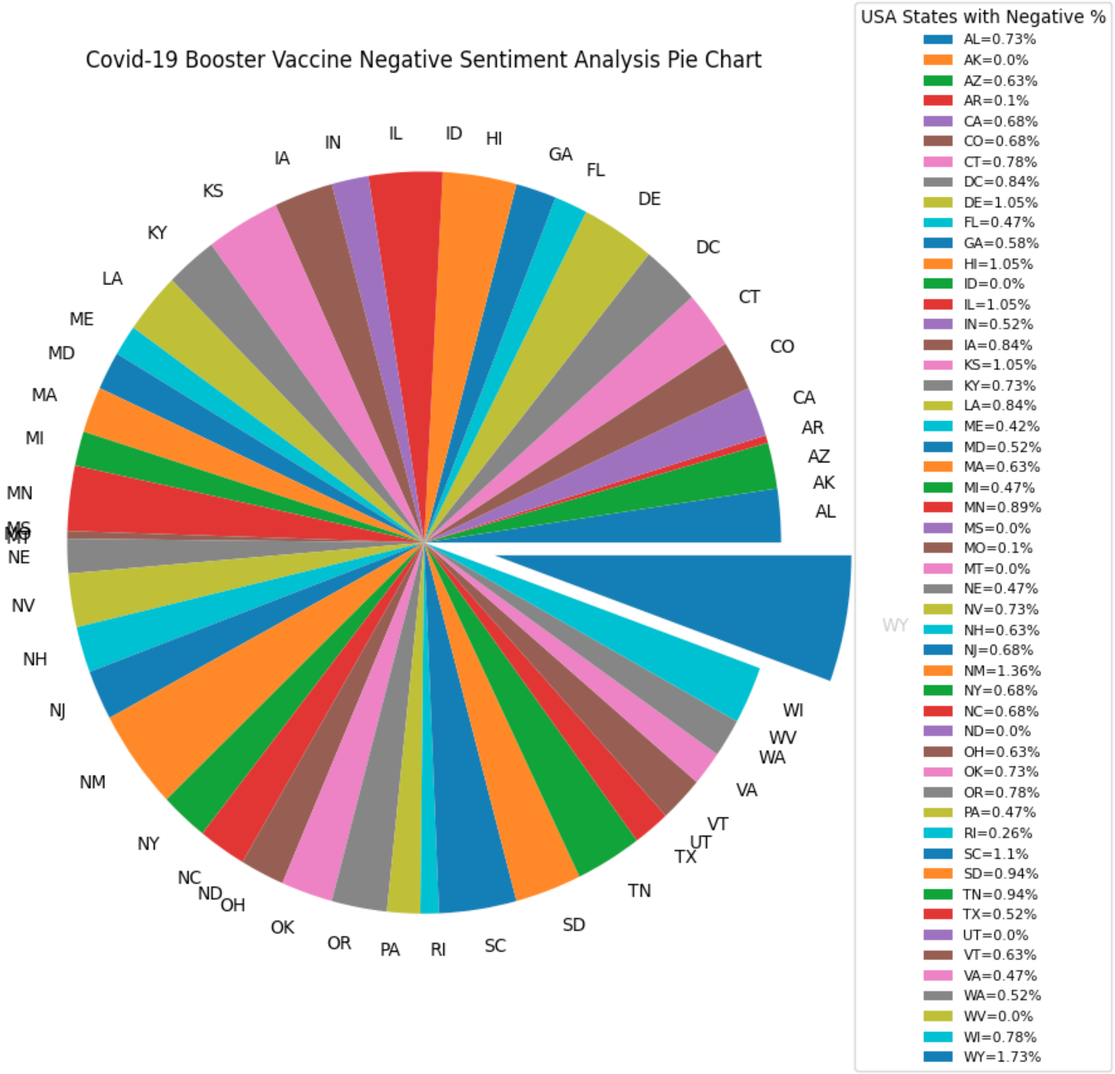


Fig 19: Pie Chart depicting negative sentiment percentage per State

From the above pie chart, we can see that the negative sentiment rate for Wyoming is the highest with 1.73% among all the 50 states in the United States of America. This implies that there is a lot of negativities about Booster Vaccine shot as analyzed by the tweet sentiment.

Word Cloud Visualizations: The magnitude of each word represents its frequency or relevance in a word cloud, which is a data visualization tool for visualizing text data. A word cloud can be used to emphasize important textual data points. Data from social networking websites is frequently analyzed using word clouds. Matplotlib, pandas, and word cloud are the modules used to create a word cloud in Python. The three-word clouds below visually depict which terms have the most frequency in our dataset.

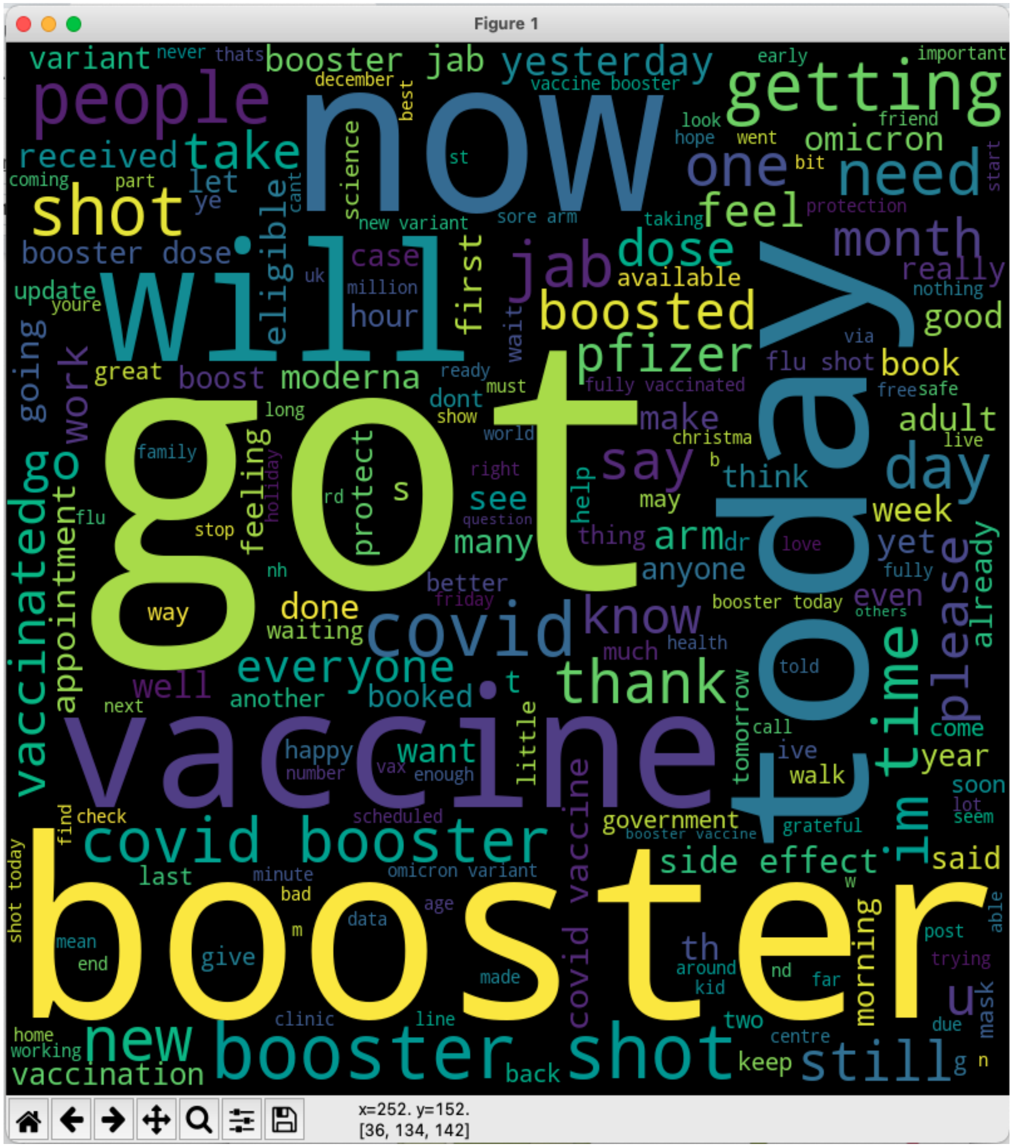


Fig 20: Word Cloud visualization of our General bucket dataset

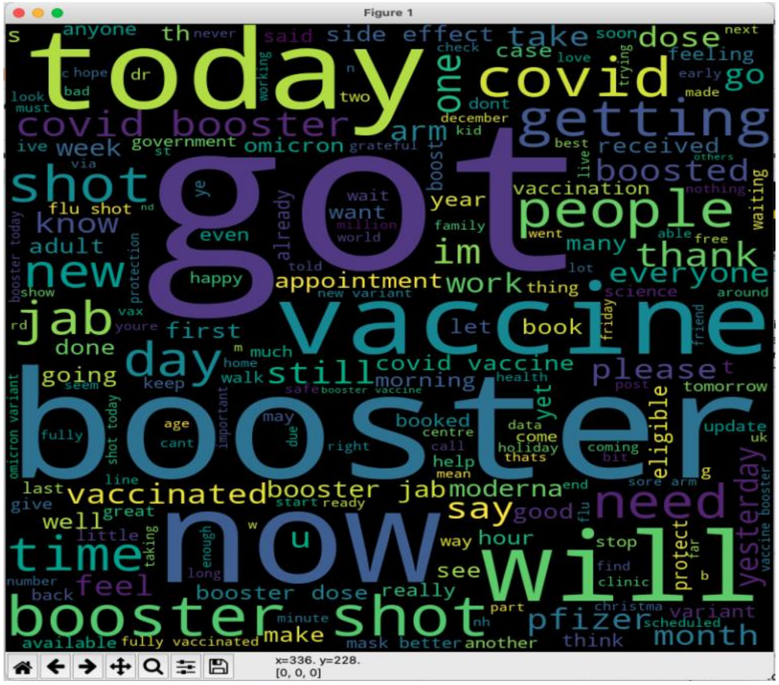


Fig 21: Word Cloud visualization of our dataset for Pfizer Vaccine

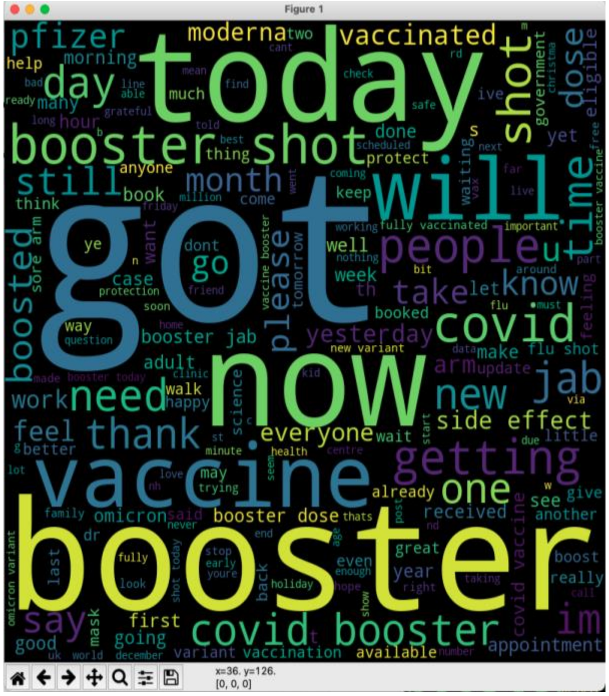


Fig 22: Word Cloud visualization of our dataset for Moderna Vaccine

2.6 Conclusion and Future Scope

We collected more than 30,000 tweets for three categories of hashtags classified as General – not specific to a vaccine, could be any vaccine: PfizerVaccine – tweets specific to Pfizer Vaccine and Modern Vaccine – tweets specific to Moderna Vaccine. These were collected using the Twitter API for the purpose of sentiment analysis. Covid-19 booster attitudes were identified from tweets, allowing for more educated judgments to be made about how to handle the present issue. We analyzed user tweets to categorize them based on tweets about COVID-19 Boosters that have been published on Twitter. This will also help researchers figure out what's keeping individuals who are eligible but unwilling to get booster shots from getting them, so they can take efforts to attain herd immunity. While most of the public has good feelings about these vaccinations, there are also negative feelings about them, according to the research. We collected tweets from all over the world, but the data was scattered for other parts of the world than the US. The future improvement of this project could be collecting the data from across the globe so that it can be visualized better.

2.7 References

1. Natural Language Toolkit Official Documentation: https://www.nltk.org/

2. Chapter 9. Twitter Cookbook, Mining the Social Web, 2nd Edition by Matthew A. Russell

3. Pandas for Data Analysis - https://pandas.pydata.org/docs/

4. Official Twitter API Documentation - https://developer.twitter.com/en/docs/twitter-api

5. Luciano Barbosa and Junlan Feng. 2010. "Robust sentiment detection on twitter from biased and noisy data", proceedings of the 23rd International Conference on Computational Linguistics: Posters, pages 36–44

3. Reflection

As part of the MSIT program, I chose to work on a group project related to “Sentiment Analysis on Covid 19 Vaccines Booster Doses using Twitter API” in the spring semester of 2022. The motivation for choosing this project was to analyze the public opinion and sentiment about Covid 19 vaccines booster doses using data collected from Twitter. Our group consisted of five members, including Priyanka, Varun Jain, Gautham, Spandana, and me. We divided the work among ourselves, with me primarily working on data cleaning and processing, and building and training the machine learning model. This project made use of the information and abilities learned during a few MSIT program classes, including Data Mining, Machine Learning, and Data Visualization. The classes gave students a foundation in data mining and machine learning methods as well as the data visualization skills necessary for presenting the sentiment analysis model's results. This project also gave me good understanding on how to extract and clean the data from end-to-end. I had the opportunity to learn and put sentiment analysis techniques to use by taking part in the Sentiment Analysis on Covid 19 Vaccines Booster Doses using the Twitter Application Programming Interface (API) project.

My activities and role in this project

As part of achieving the project objectives, technical knowledge was essential. I contributed to the project's data collection, preprocessing, sentiment analysis model testing. I also took part in team meetings to generate suggestions, offer my opinions and make sure the project was moving forward. I used my technical proficiency with the Python programming language, NLP methodologies, and other tools to complete the project's goals. I also used the project management skills to divide the effort into stories and tasks in sprints. This project is developed as part of the course ITIS 6112 Software System Design and Implementation which I took during my first semester of master’s course work which really helped me to learn the fundamental concepts of the software development life cycle process and to design and develop a sentiment analysis model using machine learning methods. I have found these concepts very much useful throughout my master’s program as these steps are to be considered and implemented for designing and developing of any web application. I have utilized the concepts from the course ITIS 6120 Applied Databases to create tables and run SQL queries. Also, I have implemented the concepts of visualization techniques from ITCS Visual Analytics.

New Avenues of Inquiry

This project highlighted the importance of data quality and the need for robust preprocessing techniques to improve the accuracy of sentiment analysis models. To effectively present the sentiment analysis results to customers, the project raised my interest in researching further data visualization techniques. To increase the scope of data collecting for sentiment analysis, it additionally sparked my interest in investigating change data sources like forums and other social media web pages. Overall, this project provided an excellent opportunity to gain hands-on experience in sentiment analysis and NLP techniques. It also highlighted the importance of working collaboratively in a team setting and the need for effective communication skills. The project's challenges provided opportunities for growth and development, and the learning objectives achieved will undoubtedly be useful in future projects.

Goals and accomplishments

The main goal of the study was to develop experience in sentiment analysis and apply NLP methods to understand the general public's views of the booster doses of the Covid 19 vaccination. I acquired expertise about how to preprocess data using the Python programming language, analyze it, and develop a sentiment analysis model using machine learning methods. Additionally, through collaborating with team members to complete the project's objectives, I improved my capacity for interaction and collaboration.

3.1 Challenges

I had faced few challenges while developing the project and they are listed below.

- First major challenge is with the data preparation stage, was to collect the data. There was secure connection issue and unable to retrieve credentials. We learnt to overcome these mistakes and we were able to collect over 30k tweets from Twitter using the provided code and the Twitter API. Thus, the technique of extracting data from data sources is frequently used for pre-processing or storage. It's crucial to obtain high-quality data. After having discussions with the professor and referencing video tutorials I was able to get the data by using preprocessing method. I had to spend enough time to get the basic data knowledge and started implementing.

- One significant challenge I faced during the project was the limited access to historical data on Twitter. I also encountered issues with data quality, as the data collected had a lot of noise and irrelevant information. These challenges required me to use robust preprocessing techniques to clean the data.
- Another challenge was when trying to remove the missing data. To successfully delete the missing data, it was necessary to additionally remove the rows that included the missing data. In this step, we eliminate any rows that contain tweets with blank text fields. We use the pandas method dropna() for this. The operation is carried out in place since "inplace" has been set to True.The existing index of the dataframe is then removed and replaced with an index of ascending integers. The reset index () method from the pandas' package is used to accomplish this.

3.2 Learnings

I acquired a lot of knowledge through my coursework, which has helped me develop into an end-to-end developer who is capable of handling everything from collecting data to data analysis in this project. I had the chance to improve my knowledge of the basic principles in excel, tableau, which was needed to design the API, and word clouds, which can be used to create visuals to evaluate functionality. I also learned about how to do data cleaning and preprocessing work which is a crucial stage in the preparation of data for sentiment analysis. It has a direct impact on a project's success. If there are missing attributes, or missing attribute values, noise, or anomalies, and redundant or incorrect data, the data is unclean. One of the most important lessons from the project was to check our work into the git repository, which serves as a central location for all our code updates. In addition to learning technical abilities, I gained teamwork techniques for finishing the project. Working in a group was a positive experience that improved my ability to manage my time, share various points of view, and take part in and appreciate each other ideas.

ssAfter finishing the entire gathering and interpreting the data part, I became more intrigued by the Data analyst role. In this position, I hope to be able to start working on the data from the requirements collecting stage to analyzing, validation, and release. The whole experience helped me a lot in successfully completing my master's degree with a strong focus in data analytics and to become Data scientist.