

FINAL PROJECT REPORT

July 2020

Project ID: SPS_PRO_1674

Project Title: Sentiment Analysis of Covid-19 Tweets – Visualization Dashboard

Start Date: 06/08/2020

End Date: 07/15/2020

Project Home Page: <https://covid-19-ibm.herokuapp.com/>

INTRODUCTION

Overview

On an average of 3-4 million tweets are shared on Twitter daily from across the country, India. As per the stats of April 2020, India comes eighth in the world with a total number of 13.15 million Twitter users. During this COVID-19 lockdown in India, people have used several social media platforms to express their feelings and share their thoughts with the world.

In this project, we have extracted country-wide, spatial Twitter data, regarding COVID-19, from India, and have analyzed millions of tweets to perform sentiment analysis a.k.a opinion mining to learn about people's sentiments during the pandemic-struck phase. The goal of our sentiment analysis project involved classifying the tweets into 'positive', 'negative', or 'neutral' polarity and presenting the data in a lucid format.

As the project's output, we have developed a sentiment analysis dashboard to visualize the polarity data.

Purpose

Today, social media covers a huge part of everyone's life. They are increasingly becoming the platform of communication for every means. Businesses can effectively utilize this by carefully listening and monitoring consumers. To properly understand customer needs, it is imperative to leverage Sentiment Analysis. Also, it can be used proactively to solve many business problems. It can also benefit Health Professionals, Policymakers, State and Central governments, and societal representatives.

LITERATURE SURVEY

Existing Problem

On 11th March 2020, the World Health Organization announced the COVID19 outbreak as a pandemic. Starting from China, this virus has infected and killed thousands of people from Italy, Spain, the USA, Iran, and other European countries as well. While this pandemic has continued to affect the lives of millions, many countries have resorted to complete lockdown. People started feeling as if they were chained, depression took over therefore, people clung to various social media

applications to share their feelings and how did they spend time at home while doing various things and keeping each other's morale up.

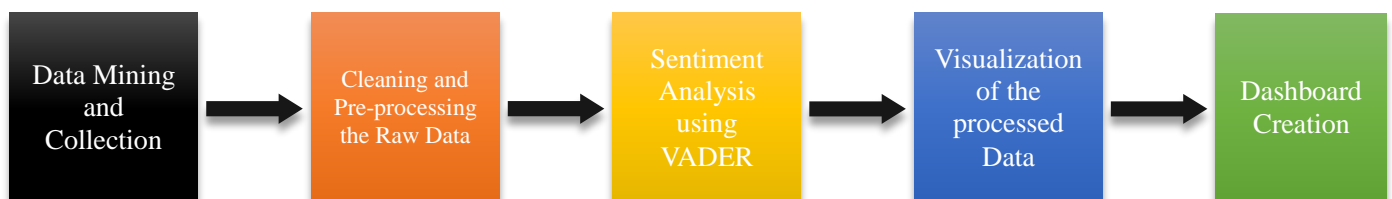
Proposed Solution

Using various Twitter and Python API's we fetched Tweets from across the country and performed Sentiment Analysis on the Tweets of the people of India to gain a wider public opinion on how the mass was keeping up during this period and then the various government and private organizations can come up with new ideas and products boost up the morale of the public.

We have used VADER (Valence Aware Dictionary and Sentiment Reasoner), a rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. VADER uses a list of lexical features that are generally labeled according to their semantic orientation as either positive or negative or neutral. It has been found successful when dealing with social media texts, editorials, reviews, etc. It not only tells us about positivity or negativity of a text but also tells us about how positive or negative a sentiment is.

THEORETICAL ANALYSIS

Block Diagram



Hardware/Software Designing

The project was built using the following hardware and software specifications:

Software:

- Python version 3.8

Hardware:

- Memory: 8 GB
- Hard Disk: 1 TB
- Processor: Intel Core i5 7th Gen

Project Specifications

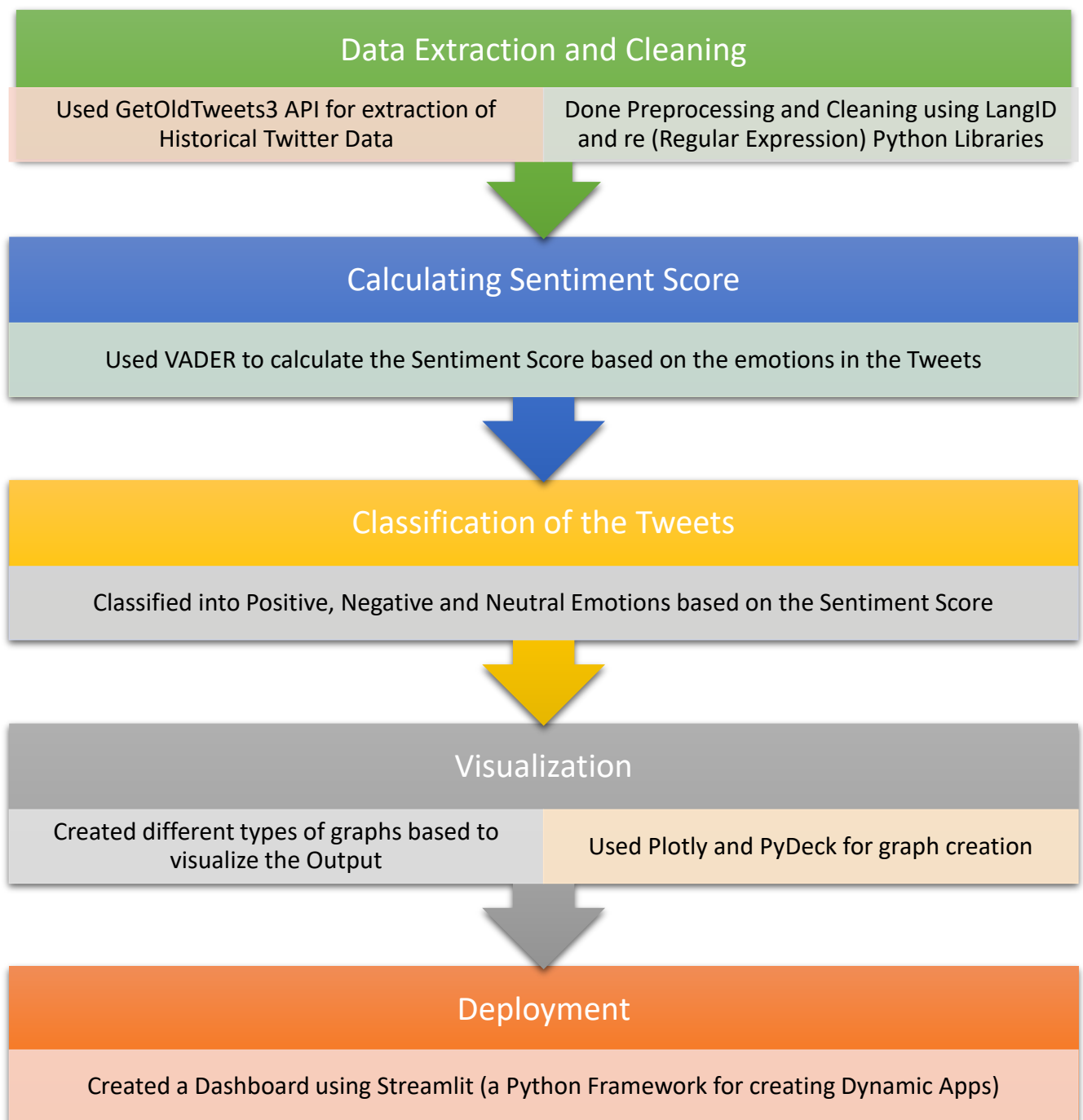
The project has been deployed on a Web server so any device that can support a Web browser with HTML5 support will be able to run the app.

EXPERIMENTAL INVESTIGATIONS

In the process of developing this Twitter Sentiment Analysis Project we have undergone a lot of brainstorming and explored various new concepts about Natural Language Processing. Experimental Investigations conducted during the process of creating the Project were on the following topics:

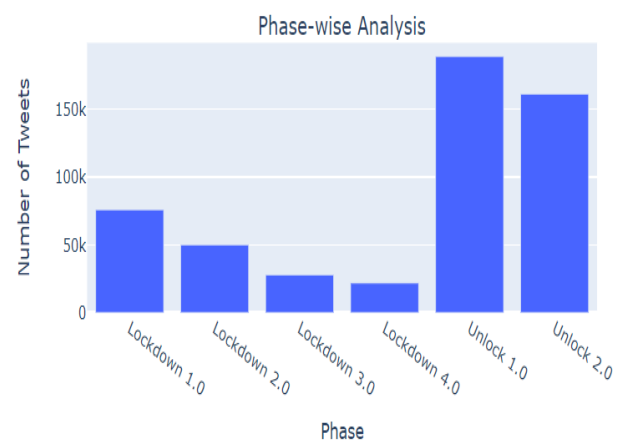
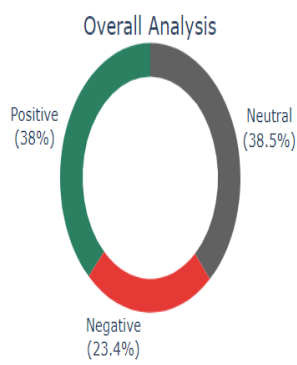
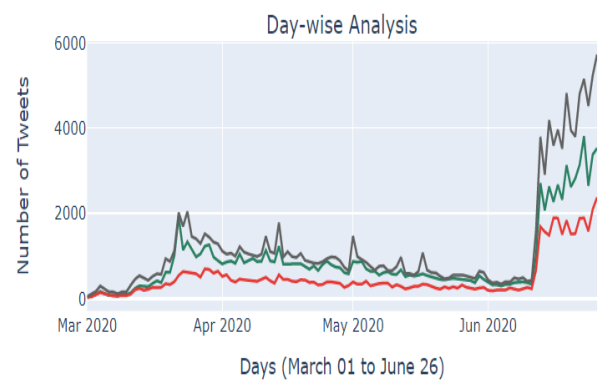
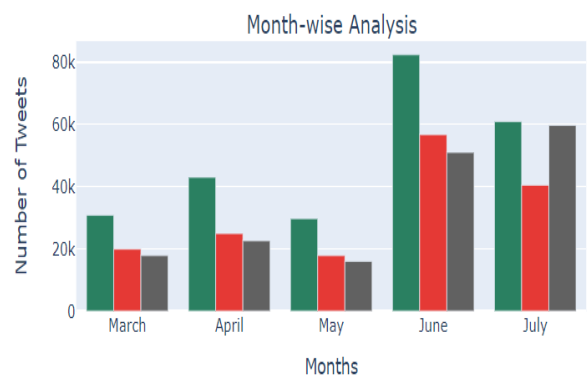
- Collection of Historical Twitter Data.
- Choosing the best model to perform Sentiment Analysis on the collected data.
- Gathering Insights from the results of the model and then Visualizations of the Outputs.
- Choosing the best Python Framework to create an app in the lowest possible size.
- Choosing the best and most dynamic platform to deploy our Visualization Dashboard.

FLOWCHART




RESULT

Dashboard



Sidebars



Pick a page

Home ▼

Authors

Animesh Singh

B.Tech, IT, 3rd year - KIET Group of Institutions

Aaditya Kapoor

B.Tech, CSE, 4th year - Galgotias University



Pick a page

Plots|



Pick plot category

Time Series plots



Pick theme

Theme



Light



Dark



Pick a page

About



About

What is



IBM HACK Challenge?



Sentiment Analysis?

Authors

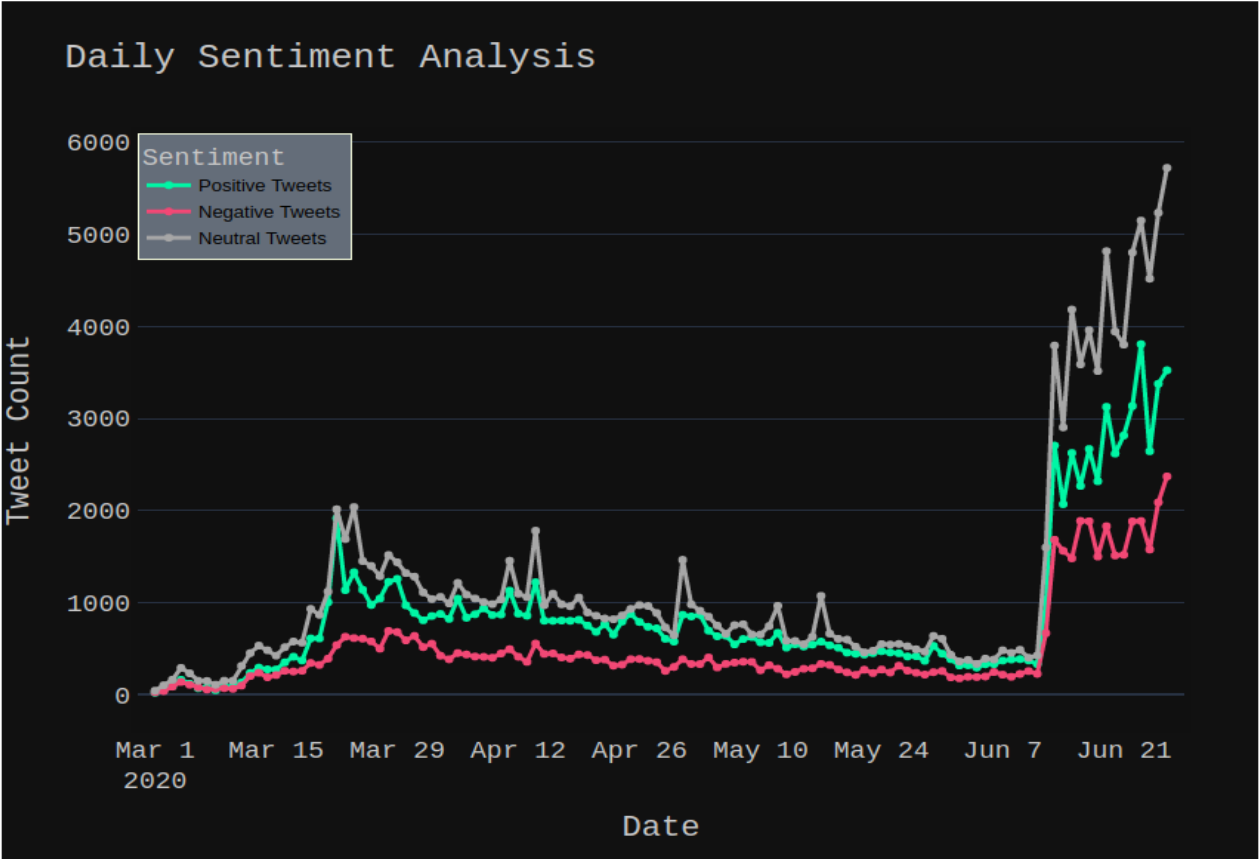
Animesh Singh

B.Tech, IT, 3rd year - KIET Group of
Institutions

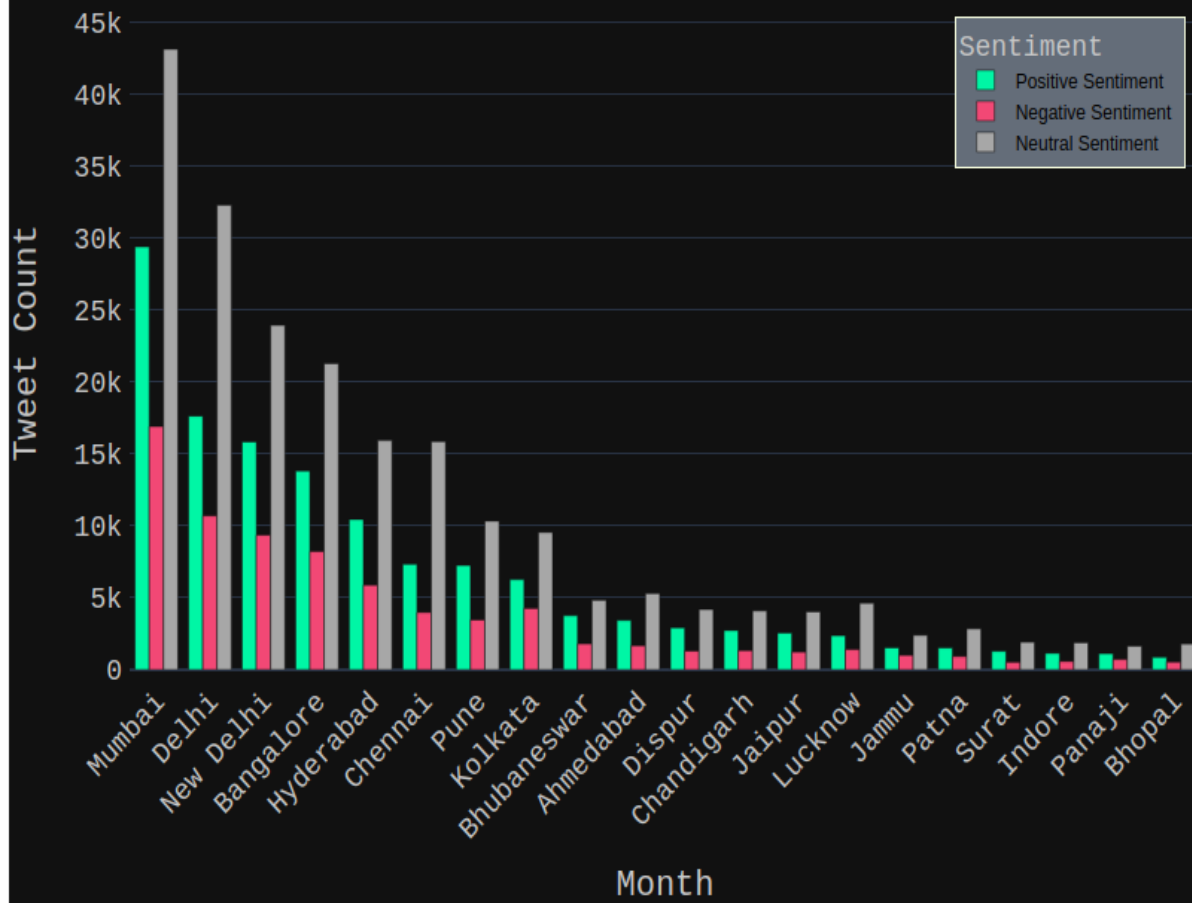
Aaditya Kapoor

B.Tech, CSE, 4th year - Galgotias
University

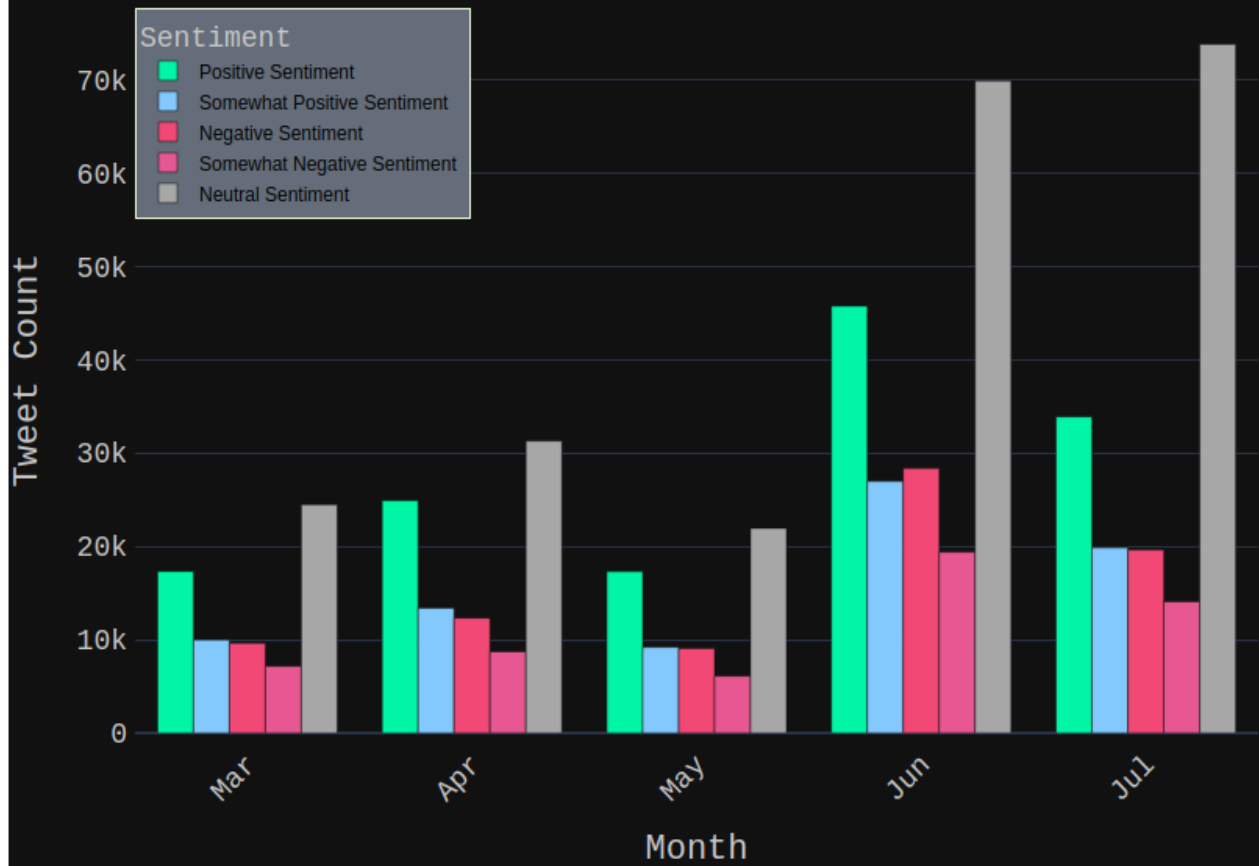
Graphs



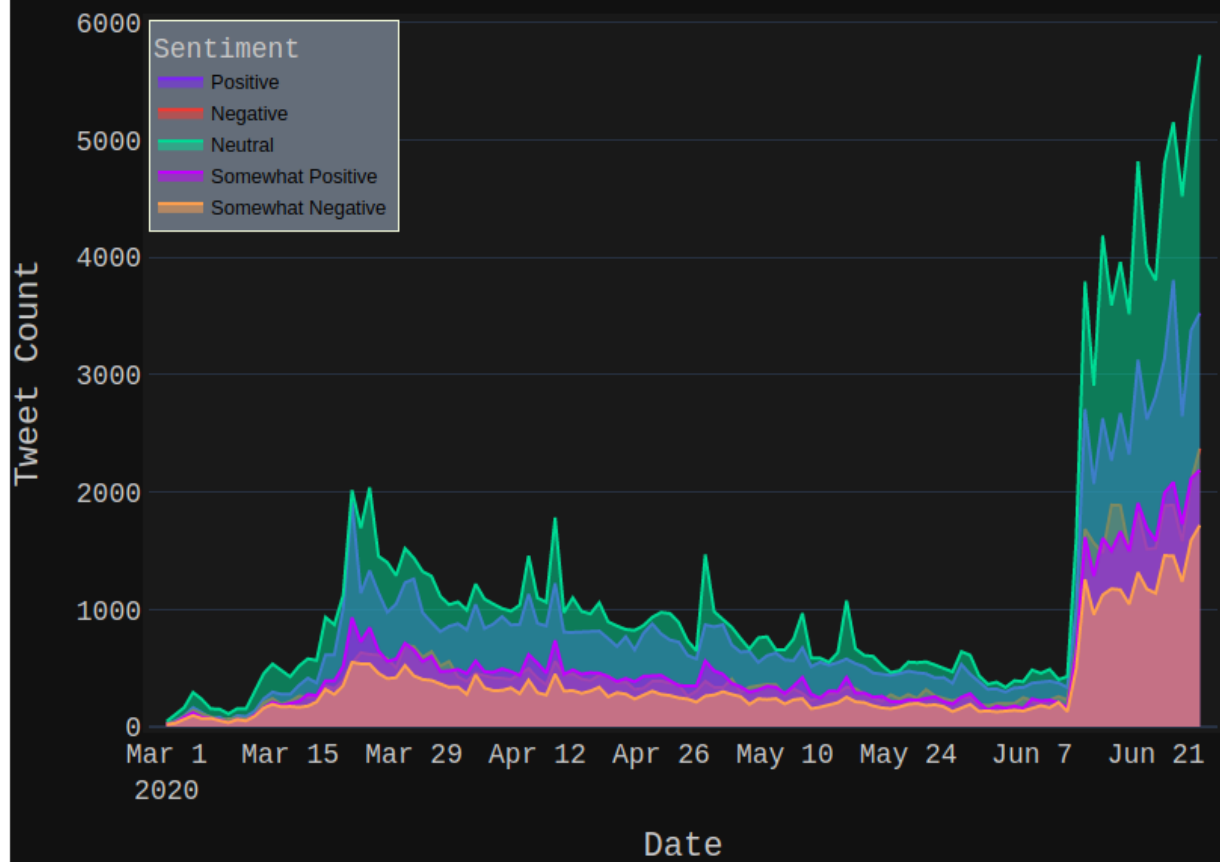
City Wise Sentiment Analysis



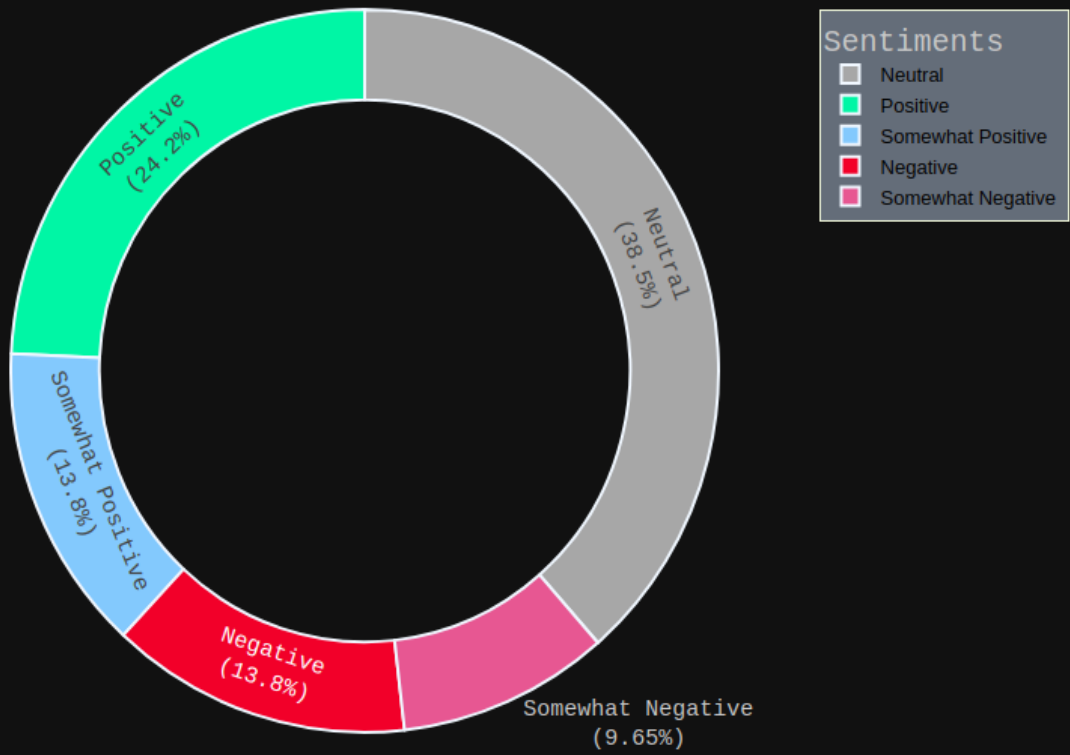
Monthly Sentiment Analysis

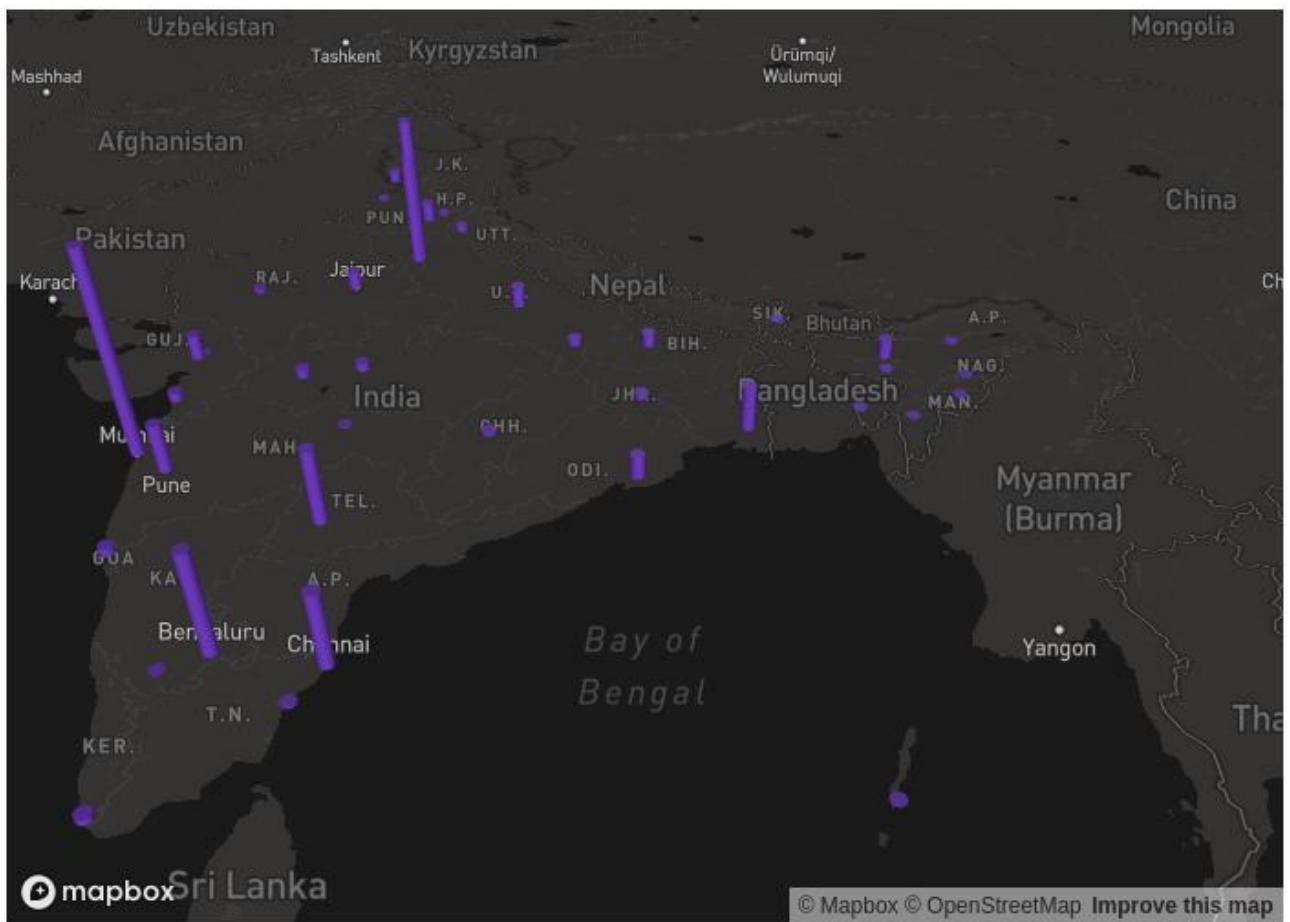


Sentiment Comparison

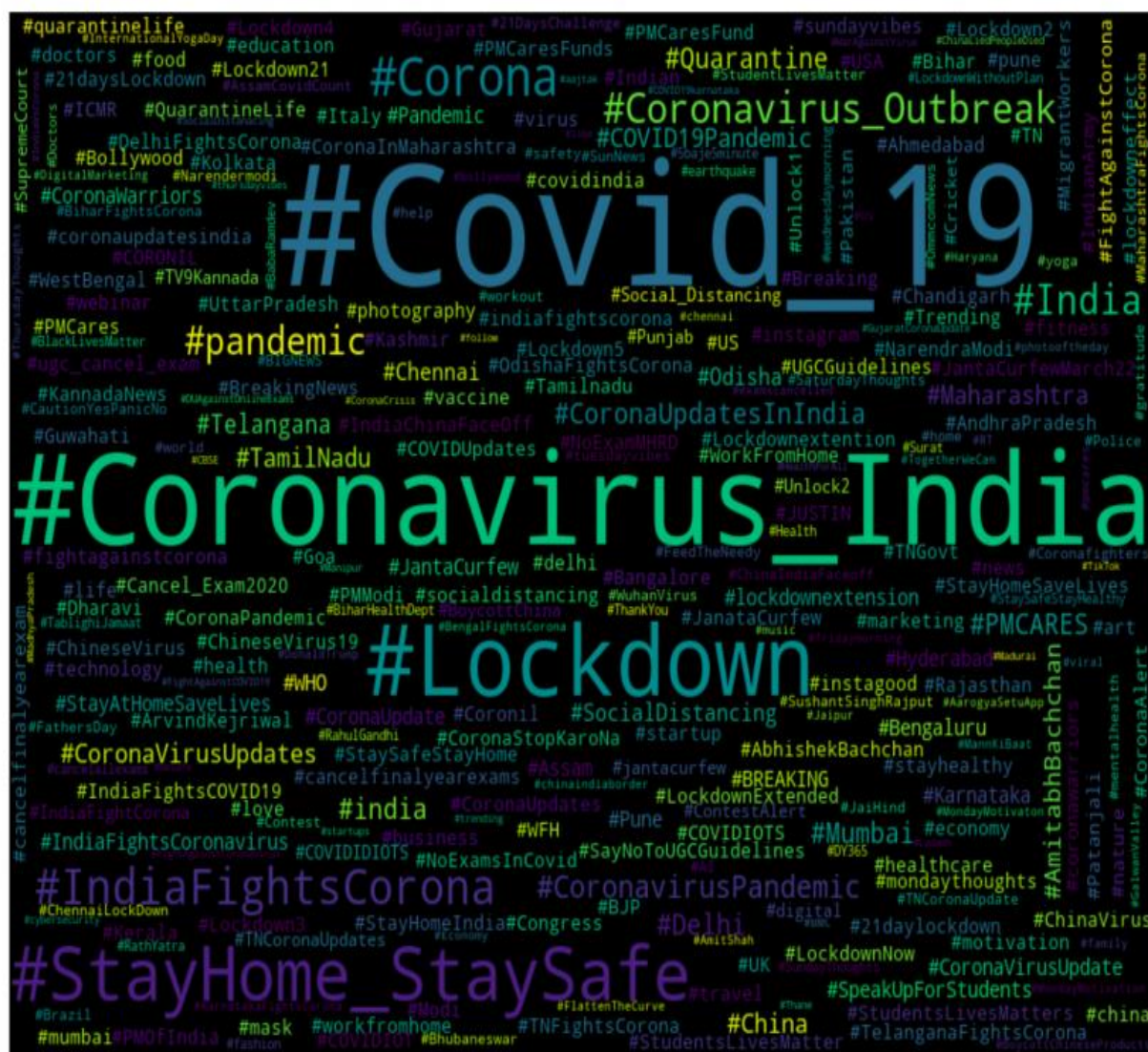


Overall Sentiment Distribution

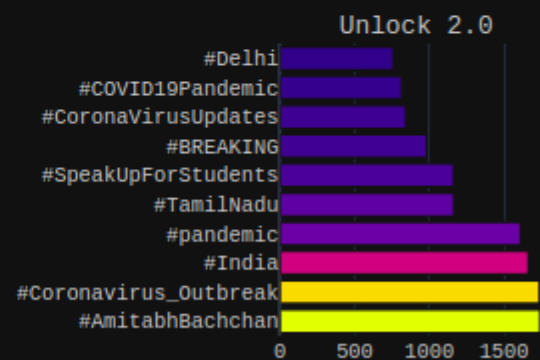
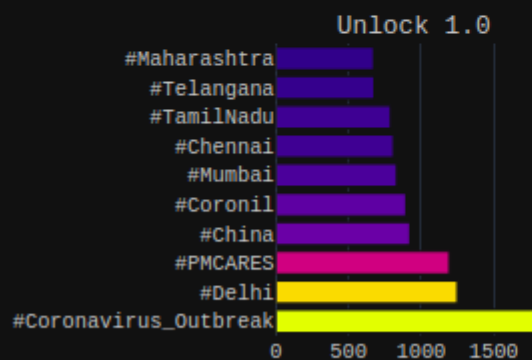
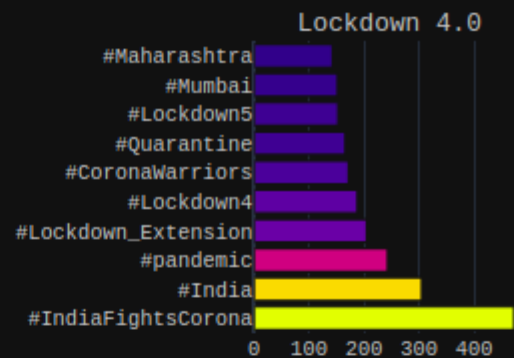
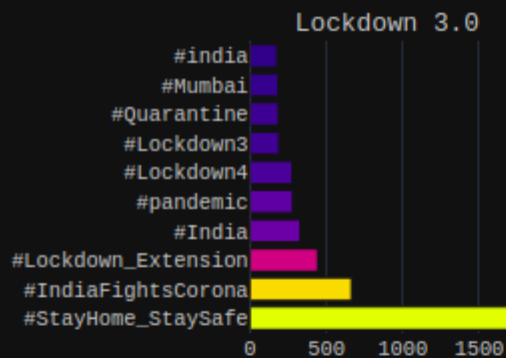
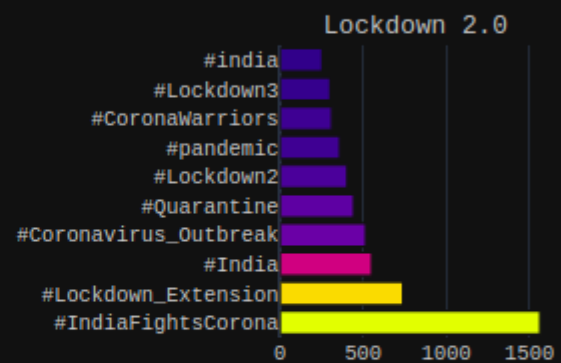
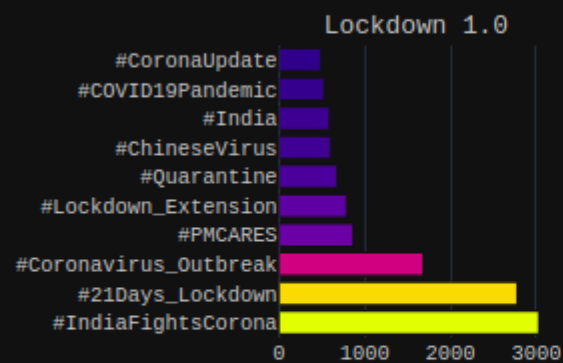




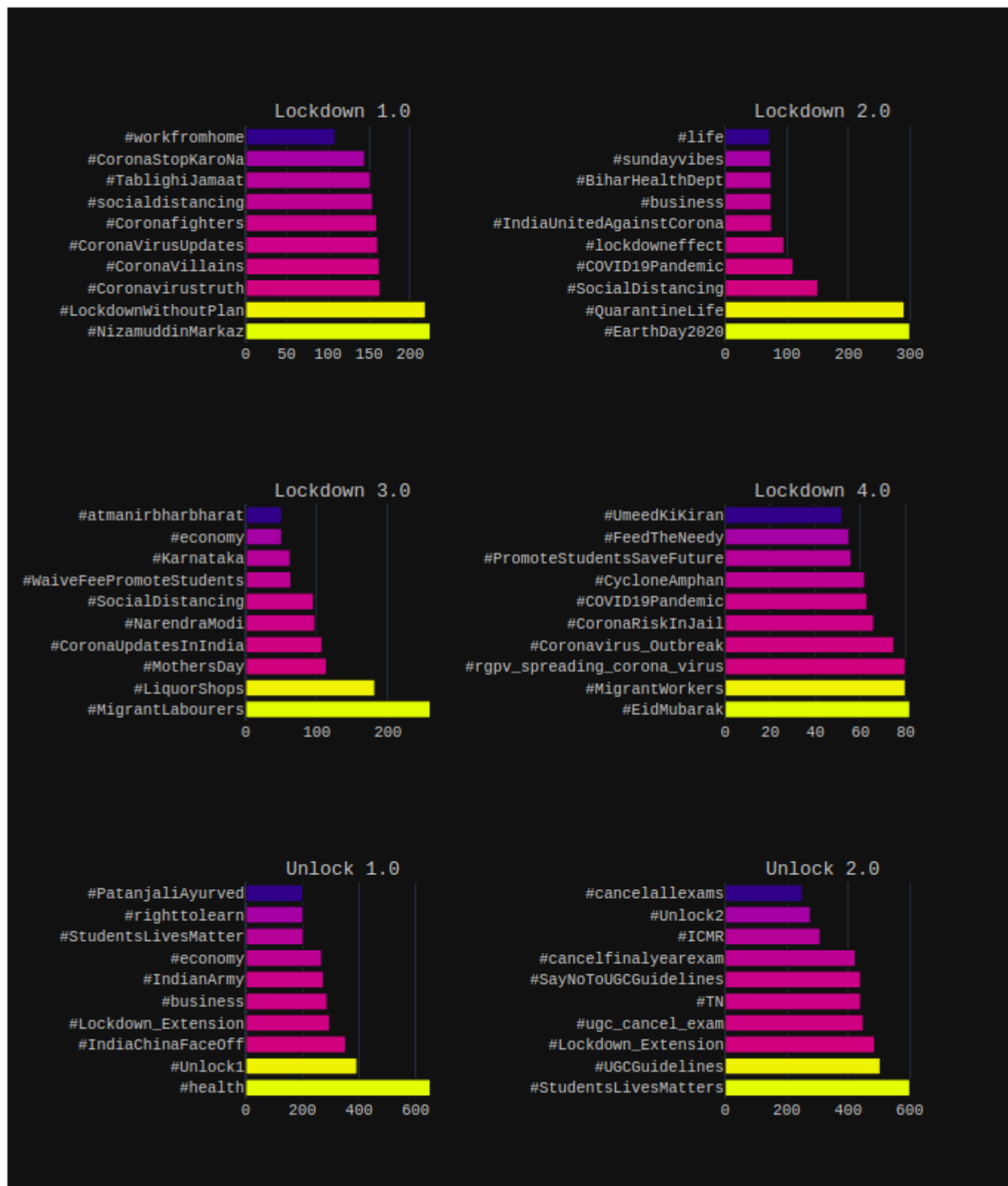
WordCloud for Covid-19 Tweets Hash-Tags



Popular Hash-Tags Analysis



Less Popular Hash-Tags Analysis



IBM HACK Challenge 2020

Covid-19 Tweet Visualization Dashboard

[About Sentiment Analysis](#)

Sentiment Analysis (a.k.a Opinion Mining)

Sentiment Analysis is a technique widely used in Data analysis. Twitter Sentiment Analysis, therefore means, using data analysis techniques to analyze the sentiment of the tweets, thus classifying tweets into the categories of positive, negative and neutral.

In this project, we have used the VADER Sentiment Analysis tool for classifying tweets into positive, negative or neutral polarity. Below, is a text field in which you can enter any sentence to find out its polarity to see sentiment classification in action!!

Enter text below

I am super excited today!!

Sentiment output = 0.7835

Positive text predicted

A positive value above means the entered text is classified as a *positive* text. Similarly, a negative value means the text entered is *negative* and a value close to 0 means that the text is *neutral*.

IBM HACK Challenge 2020

Covid-19 Tweet Visualization Dashboard

[About IBM HACK Challenge](#)

CODE FOR A BRIGHTER TOMORROW

IBM Hack Challenge 2020 is all about *coding for a cause*. It's about finding solutions to problems plaguing society today. It is an opportunity to showcase your coding talent, learn new technologies and build an efficiently working solution.

[Our problem statement](#)

Sentiment Analysis of COVID-19 Tweets Visualization Dashboard

The sentiment analysis of Indians after the extension of lockdown announcements to be analyzed with the relevant #tags on twitter and build a predictive analytics model to understand the behavior of people if the lockdown is further extended. Also develop a dashboard with visualization of people reaction to the govt announcements on lockdown extension

ADVANTAGES & DISADVANTAGES

Some Advantages:

- **Works exceedingly well on social media** type text such as public opinion.
- **Doesn't require any training data** but is constructed from a generalizable, valence-based, human-curated gold standard sentiment lexicon.
- **Fast** enough to be used online with streaming data.
- Does not severely suffer from a **speed-performance trade-off**.
- **Efficient** at analysing large datasets.
- Can identify whether a phrase is positive, negative or neutral, which in turn can be used to determine a customer's sentiment towards a brand or service. This helps the businesses to identify their strengths and weaknesses.
- It can also help to identify marketing campaigns that are not working well.

Some disadvantages:

- Spellings and grammatical mistakes may cause the analysis to overlook important words or usage.
- Sarcasm and irony may be misinterpreted.
- Analysis is language-specific.
- Discriminating jargon, nomenclature, memes, or turns of phrase may not be recognized.
- Unavailability of Twitter Historical Data.

APPLICATIONS

This automated machine learning driven sentiment analysis could help the following people and industries:

- Health Professionals.
- Policy makers.
- State and Central Governments to understand and identify rapidly changing psychological risks in the population.
- Timely responses and initiatives taken by the agencies to mitigate and prevent adverse emotional and psychological consequences will significantly improve public crisis and phenomenon.
- Provide valuable insights on attitudes, perceptions and behaviours for critical decision making for business, political leaders and societal representatives.
- Corporations and small businesses can also benefit through such analyses and machine learning models to better understand consumer sentiment and expectations.

CONCLUSION

This project deals with the sentiment analysis of Indians a few days before and after the lockdown announcements were made. We used the social media platform Twitter for our analysis. Tweets

were studied to gauge the feelings of Indians towards the lockdown. Tweets were extracted using the following prominent hashtags namely: #COVID, #Coronavirus, #Lockdown, #Pandemic, and #PMCare from March 1st to July 10th, 2020. A total of 5, 74,108 tweets were considered for the analysis. The analysis was done using Python and different graphs were generated that depicts the sentiments of the tweets.

Overall, it can be seen that Indians have taken the fight against COVID19 positively and the majority are in agreement with the government for announcing the lockdown to flatten the curve. It could be seen from the tweets that several people were angry that the lockdown came a bit late. It should have been announced a week prior. Also, some tweets expressed concerns that the passengers from abroad who flew in should have been quarantined before letting them reunite with their families. Nevertheless, as of now, the lockdown response seems positive and indicates that India has succeeded in controlling the coronavirus spread to a great extent.

FUTURE SCOPE

Future studies can look into pre and post lockdown tweets and understand whether there was a change in sentiments from the beginning to the end of the lockdown. Also, future studies can look into factors that affect mental health during lockdowns and pandemic spreads. Another area for future research could be tackling of fake news that gets circulated through social media, impacting the mental health of the receivers. Many private and public industries can also research to create products for the post lockdown situation of the country

BIBLIOGRAPHY

Names: Animesh Singh and Aaditya Kapoor

College Names: KIET Group of Institutions and Galgotias University

Work Title: Sentiment Analysis of Covid-19 Tweets – Visualization Dashboard

References:

Research and Feasibility Study for Twitter Sentiment Analysis

- <https://www.saifmohammad.com/WebDocs/emotion-survey.pdf>
- <https://github.com/abdufatir/twitter-sentiment-analysis>
- <https://github.com/cjhutto/vaderSentiment>
- <https://github.com/sloria/textblob>
- <https://cloud.ibm.com/apidocs/tone-analyzer?code=python>
- <https://medium.com/@Intellica.AI/vader-ibm-watson-or-textblob-which-is-better-for-unsupervised-sentiment-analysis-db4143a39445>
- <https://towardsdatascience.com/twitter-sentiment-analysis-based-on-news-topics-during-covid-19-c3d738005b55>
- <https://python.gotrained.com/tf-idf-twitter-sentiment-analysis/>
- <https://www.kaggle.com/satanizer/covid-19-tweets-analysis>

Gathered Twitter Data using this Python Library

- <https://github.com/Jefferson-Henrique/GetOldTweets-python>
- <https://github.com/Mottl/GetOldTweets3>

Python Libraries and Frameworks used

- <https://pypi.org/project/vaderSentiment/>

- [https://plotly.com/python/getting-started/#:~:text=Plotly%20in%20Python-,Overview,the%20Plotly%20JavaScript%20library%20\(plotly.](https://plotly.com/python/getting-started/#:~:text=Plotly%20in%20Python-,Overview,the%20Plotly%20JavaScript%20library%20(plotly.)
- <https://pypi.org/project/pydeck/>

Python Dynamic App Creation

- <https://www.streamlit.io/>

Deployment Platform

- <https://www.heroku.com/>

APPENDIX

Source Code

Data Extraction

Importing Python Libraries

```
from datetime import date, timedelta
import GetOldTweets3 as got
import time
import pandas as pd
```

Start and End Date for Data Extraction

```
sdate = date(2020, 6, 26)
edate = date(2020, 6, 27)
```

```
delta = edate - sdate
```

Cities of which Data has been Collected

```
cities1 = ['Mumbai', 'Delhi', 'Bangalore', 'Chennai', 'Kolkata', 'Ahmedabad', 'Jaipur', 'Chandigarh',
'Lucknow', 'Varanasi', 'Panaji', 'Jammu', 'Gandhinagar', 'Gangtok', 'Aizawl', 'Amravati', 'Itanagar',
'Dispur', 'Patna', 'Shimla', 'Ranchi', 'Bengaluru', 'Thiruvananthapuram', 'Surat', 'Jodhpur', 'Bhopal',
'Indore', 'Pune', 'Imphal', 'Bhubaneswar', 'Hyderabad', 'Mysore', 'Dehradun', 'Port Blair']
```

```
cities2 = ['Daman & Diu', 'Raipur', 'New
Delhi', 'Lakshadweep', 'Shillong', 'Kohima', 'Agartala', 'Pondicherry', 'Amritsar']
```

Data Extraction has been done using these Hashtags

```
tags = ['covid', 'coronavirus', 'lockdown', 'pandemic', 'PMcares']
```

```
cities = cities1+cities2
```

```

error_dates = []
error_cities = []
error_tags = []
tweet_data = []

for i in range(0, delta.days + 1,1):
    day_s = str(sdate + timedelta(days=i))
    day_e = str(sdate + timedelta(days=i+1))

    for tag in tags:
        for city in cities:
            tweetCriteria = got.manager.TweetCriteria().setQuerySearch(tag)\
                .setSince(day_s)\
                .setUntil(day_e)\
                .setNear(city)

# TRY CATCH BLOCK to avoid any kind of errors while extracting data
    try:
        print("Searching for tweets...\tCity-> {} \tDate-> From {} to {}".format(city, day_s,
day_e))
        tweets = got.manager.TweetManager.getTweets(tweetCriteria)
        print("Search complete!!")

    except:
        print("\n\nError occurred..Going to sleep for 14 minutes.")
        error_dates.append([day_s, day_e])
        error_cities.append(city)
        error_tags.append(tag)

        # Sleeping for 10 minutes to reset limit

        time.sleep(14*60)
        print("\n\nWaking up..t*yawn*")

        # Skipping this iteration to avoid storing redundant tweets
        # in tweet_data

        continue

    for tweet in tweets:

tweet_data.append([tweet.text,tweet.date,tweet.retweets,tweet.favorites,tweet.hashtags,city])
        print("Data saved!")
        print("\n\nTweets so far: {}".format(len(tweet_data)))

if len(error_cities) > 0:
    print("\n\nNOTE: Tweets were missed for: {} cities on these dates: {}".format(error_cities,
error_dates))

```

else:

```
print("No errors occurred during execution.Proceed to creating a dataframe and save it in your drive!")
```

```
print("Making dataframe..")
```

```
df = pd.DataFrame(tweet_data, columns=['Text', 'Date', 'Retweets', 'Favs', 'Hashtags', 'City'])
```

```
print("Exporting dataframe..")
```

```
df.to_csv("Tweets from '+str(sdate)+' to '+day_e+'.csv')
```

```
print("Data exported successfully!")
```

Cleaning and Preprocessing

```
import re # Importing Regular Expressions
```

```
# Function to remove URLs
```

```
def remove_urls(df, column_name):
```

```
    # This will remove all the urls from the tweets
```

```
    df[column_name] = df[column_name].apply(lambda x :
```

```
    ""'.join(re.sub(r'((www\.[\S]+)|(https?://[\S]+))', '', x)))
```

```
    return df
```

```
# Function to remove Mentions
```

```
def remove_mentions(df, column_name):
```

```
    # This will remove mentions (e.g. @elon_musk, @animesh983881 etc) from the tweets
```

```
    df[column_name] = df[column_name].apply(lambda x : ""'.join(re.sub(r'@[\S]+', '', x)))
```

```
    return df
```

```
# Function to remove Retweets
```

```
def remove_RT(df, column_name):
```

```
    # A lot of tweets will contain retweet information as RT as a tweet's prefix
```

```
    # Since, the tweets are all already converted to lower case, we are replacing 'rt' and not 'RT'
```

```
    df[column_name] = df[column_name].apply(lambda x : ""'.join(re.sub(r'\brt\b', '', x)))
```

```
    return df
```

Sentiment Analysis

```
# Importing VADER
```

```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
```

```
sa_object = SentimentIntensityAnalyzer()
```

```
scores = []
```

```
for text in df['Text']:
```

```
    score = sa_object.polarity_scores(text)['compound']
```

```
    scores.append(score)
```