



# Sentiment Analysis of COVID-19 Tweets Visualization Dashboard

<sup>1</sup>Banshri Modhiya,<sup>2</sup>Niharika Solanki,<sup>3</sup>Poonam Faldu

<sup>1</sup>Assistant professor., <sup>2</sup>Assistant professor, <sup>3</sup>Assistant professor  
Computer Engineering department,  
Parul University, Vadodara

## Abstract

Microblogging today has become a very popular communication tool among Internet users. Millions of users share opinions on different aspects of life every day. Therefore microblogging web-sites are rich sources of data for opinion mining and sentiment analysis. Because microblogging has appeared relatively recently, there are a few research works that were devoted to this topic. In our paper, we focus on using Twitter, the most popular microblogging platform, our purpose of this paper is to introduce a new strategy to identify how the peoples are responding to the government's decision of lockdown. Crowded regions with actively moving people (called at-risk regions) are susceptible to spreading the disease, especially if they contain asymptomatic infected people together with healthy people so the government took the very important decision of Lockdown. In this paper we have proposed identifies the sentiments of the tweets with the relevant #tags of COVID. The Tweets was collected by Twitter's API then the tweets was stored in the database and then the frequency of words in the tweets was analyzed by IBM's NLP and will return the sentiment of the tweet, from that extracted sentiments we'll get to know how the citizens take the government's decision, it was also allow us to identify whether to extend lockdown or not.

**Keywords:** COVID-19, Lockdown, Sentiment.

## INTRODUCTION

Microblogging today has become a very popular communication tool among Internet users. Millions of messages are appearing daily in popular web-sites that provide services for microblogging such as Twitter<sup>[1]</sup>, Tumblr<sup>[2]</sup>, Facebook<sup>[3]</sup>. Authors of those messages write about their life, share opinions on variety of topics and discuss current issues. Because of a free format of messages and an easy accessibility of microblogging platforms, Internet users tend to shift from traditional communication tools (such as traditional blogs or mailing lists) to microblogging services<sup>[12]</sup>. As more and more users post about products and services they use, or express their political and religious views, microblogging web-sites become valuable sources of people's opinions and sentiments. Such data can be efficiently used for marketing or social studies. We use a dataset

formed of collected messages from Twitter<sup>[10]</sup>. Twitter contains a very large number of very short messages created by the users of this microblogging platform. The contents of the messages vary from personal thoughts to public statements. In this paper deals with the sentiment analysis of Indians after the lockdown announcements were made. We used Twitter for our analysis. Tweets were analyzed to get sentiments and emotions of Indians towards the lockdown and global pandemic. Tweets were extracted using the two hashtags namely: #Lockdown and #COVID-19, but here we used the longitude and latitude coordinates of India to extract Tweets, we also extract tweets from some metropolitan cities.

Sentiment analysis is also known as “opinion mining” or “emotion Artificial Intelligence” and alludes to the utilization of natural language processing (NLP), text mining, computational linguistics, and bio measurements to methodically recognize, extricate, evaluate, and examine emotional states and subjective information. Sentiment analysis is generally concerned with the voice in client materials; for example, surveys and reviews on the Web and web-based social networks. The analysis was done using the NLP (Natural Language Processing) and Python, and plot charts using that data, the charts get updated after 24hrs<sup>[12]</sup>.

From the analysis, it can be seen that the majority are in agreement with the government for announcing the lockdown to avoid a pandemic. It could be seen from the tweets that several people were neutral about the decision.

## **PROBLEM STATEMENT**

The main aim is to analyze 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 behaviour of people if the lockdown is further extended.

## **NOVELTY**

In this Paper, We have made a web-app that provides Sentiment Analysis of tweets from India. It shows results according to dates. It also shows an analysis of tweets from major cities. Apart from this, it also shows the latest tweets regarding COVID-19. And data get updated after 24hrs.

## **SOCIAL IMPACT**

With the help of this sentiment analysis, we conclude that how many people take government decisions positively or negatively.

Authorities can provide a quicker response to people according to the situation.

## **SOURCES THAT USED TO DEVELOP WEB-APP**

Python 3.8

Tweepy

IBM Watson Studio

IBM NLP Service

IBM Cloud

Django framework

HTML5 &amp; CSS3

JavaScript

Chart.js

Bootstrap

## FEATURES

**All Over Data:** This All Over Data is the home page of the web-app. This contains different graphs or charts for the results of the sentiment analysis and latest tweets regarding COVID19.

**Major Cities:** This page contains different charts or graphs of data on the sentiment analysis done location-wise. We have selected the main 4 locations (Mumbai, Delhi, Bangalore, and Chennai) and we will visualize the data according to the location.

**Responsive Design:** The design of the site is responsive i.e. it can be viewed on any device without facing any trouble.

**Sentiment checker:** In this page anyone can see the live sentiment of his/her own tweet.

**Covid Cases:** The real time graph of the covid cases in India will be shown here.

## PROPOSED WORK AND IMPLEMENTATION

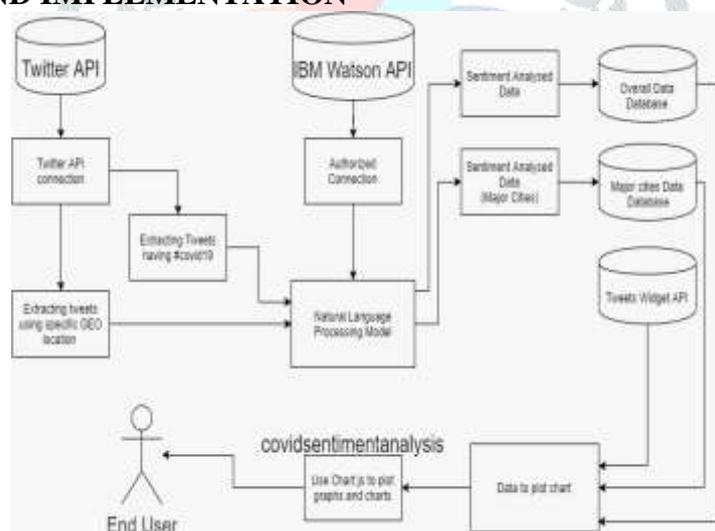


Figure:1 DFD Diagram

In this paper, we made a developer account on Twitter, by using that we got the Twitter API key. By using twitter API, we are extracting tweets having hashtags regarding COVID-19 of past 7 days.

Show the figure, we simply enter geolocation of India as we want overall tweets from India. We are creating a dictionary of cities and their geolocation (longitude, latitude, radius (km or miles)) and accessing the items of the dictionary and pass geolocation as an argument.

Then we enabled and made an Authorized connection of the IBM Watson API of Natural Language Processing.

Then by using that Natural language processing we prepared a model in which all the tweets will be analyzing the collected tweets.

All the process of tokenizing the words and removing the stop words will be done in this model.

IBM Watson's Natural Language Processing API will help us to get the Sentiments.

IBM Watson's Natural Language Processing API will return the sentiments (Positive, Negative, Neutral) and also (Joy, Angry, sad) of all the collected tweets of all over India as well as of specific Geolocations (major cities).

Then we stored the collected data on the database.

We made 2 different Databases for all over India's tweets and also for major cities.

Then we fetch the stored data from the database to visualize the plots further.

We used different types of plots such as line graph & Bar graph to show polarity and emotion.

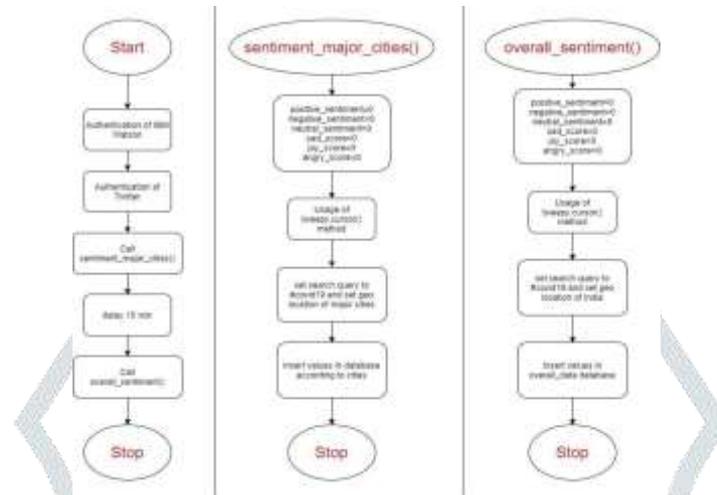


Figure: 2 .Flowchart of proposed Work

#### Step:1 TwitterAPI Connection:

By using twitter API, we are extracting tweets having hashtags regarding COVID-19. We are using geolocation to extract tweets from a specific location. As shown in the above in flowchart there are two functions in the program “sentiment\_major\_cities()” and “overall\_sentiment()”. In “sentiment\_major\_cities()” we are passing geolocation of major cities in twitter search API as shown below:

dictionary of cities and their geolocation(longitude, latitude, radius(km or miles)) and accessing the items of dictionary and pass geolocation as argument value in “tweepy.Cursor()” (refer highlighted part).

```
# dictionary of longitude,latitude,radius of cities
cities = {
    "Delhi": "28.7041,77.1025,21.73km",
    "Chennai": "13.0827,80.2707,11.65km",
    "Mumbai": "19.0760,72.8777,13.86km",
    "Banglore": "12.9716,77.5946,15.02km"
}
```

```

def sentiment_major_cities():
    city=0
    t=0
    positive_sentiment = 0
    negative_sentiment = 0
    neutral_sentiment = 0
    sad_score = 0
    joy_score = 0
    anger_score = 0
    x = datetime.datetime.today()
    i = x - datetime.timedelta(days=1)
    while (t<1):
        for a, y in cities.items():
            city+=1
            positive_sentiment = 0
            negative_sentiment = 0
            neutral_sentiment = 0
            sad_score = 0
            joy_score = 0
            anger_score = 0
            search_words = "#covid19 OR #lockdown OR #coronavirus" + " -filter:retweets"

            if city == 4:
                time.sleep(15*60)

            tweets = tweepy.Cursor(api.search,
                                   q=search_words,
                                   lang="en", geocode=y, until=x.date()).items(500)

```

And in “overall\_sentiment()” we simply enter geolocation of India as we want overall tweets from India, As shown below:

```

def overall_sentiment():
    t=0
    positive_sentiment = 0
    negative_sentiment = 0
    neutral_sentiment = 0
    sad_score = 0
    joy_score = 0
    anger_score = 0
    search_words = "#covid19 OR #lockdown OR #coronavirus" + " -filter:retweets"
    x = datetime.datetime.today()
    i = x - datetime.timedelta(days=1)
    while (t<1):
        tweets = tweepy.Cursor(api.search,
                               q=search_words,
                               lang="en", geocode="30.5937,78.9639,1023.140km", until=x.date()).items(500)

```

## Step:2 IBM Watson NLP API Connection:

After extracting tweets from a specific location, we analyzed the sentiment of tweets by using the NLP model which is provided by IBM.

```

# connection of IBM Watson Studio
authenticator = IAMAuthenticator('7upwbu580Gn2m2s3TR-9f1fjvh2JE34dKTPeR')
natural_language_understanding = NaturalLanguageUnderstandingV1(
    version='2019-07-12',
    authenticator=authenticator)

natural_language_understanding.set_service_url('https://api.eu-gb.natural-language-understanding.watson.cloud')

```

We analyze the sentiment through API calls, by just passing text of tweets.

```

for tweet in tweets:

    response = natural_language_understanding.analyze(
        text=tweet.text,
        language="en",
        features=Features(sentiment=SentimentOptions(document=True),
            emotion= EmotionOptions(document=True))).get_result()

```

### Step:3 Connection of Database and store data in Database:

```

if score > 0:
    positive_sentiment += 1
elif score < 0:
    negative_sentiment += 1
else:
    neutral_sentiment += 1

if sad_emotion > anger_emotion and sad_emotion > joy_emotion:
    sad_score +=1
if anger_emotion > sad_emotion and anger_emotion > joy_emotion:
    anger_score+=1
if joy_emotion > anger_emotion and joy_emotion > sad_emotion:
    joy_score +=1

mydb = mysql.connector.connect(
    host="covidsentimentanalysis.mysql.pythonanywhere-services.com",
    user="covidsentimentan",
    password="administrator",
    database = "covidsentimentan$tweets"
)
mycursor = mydb.cursor()

sql = "INSERT INTO overall_sentiment VALUES ( %s,%s, %s,%s,%s,%s,%s)"
val = (str(i.date()),positive_sentiment,negative_sentiment,neutral_sentiment,sad_score,anger_score,joy_score)
mycursor.execute(sql, val)
mydb.commit()

```

### Step:4 Fetching values from Database for plotting charts:

```

class Data:
    def __init__(self,loc):
        self.loc = loc.lower()

    date_data_query={
        'mumbai':"select * from mumbai_sentiment",
        'delhi':"select * from delhi_sentiment",
        'bangalore':"select * from kolkata_sentiment",
        'chennai':"select * from chennai_sentiment",
        'overall':"select * from overall_sentiment"
    }

    overall_query={
        'mumbai':"select sum(positive),sum(negative) ,sum(neutral),sum(sad),sum(anger),sum(joy) from mumbai_sentiment",
        'delhi':"select sum(positive),sum(negative) ,sum(neutral),sum(sad),sum(anger),sum(joy) from delhi_sentiment",
        'bangalore':"select sum(positive),sum(negative) ,sum(neutral),sum(sad),sum(anger),sum(joy) from kolkata_sentiment",
        'chennai':"select sum(positive),sum(negative) ,sum(neutral),sum(sad),sum(anger),sum(joy) from chennai_sentiment",
        'overall':"select sum(positive),sum(negative) ,sum(neutral),sum(sad),sum(anger),sum(joy) from overall_sentiment"
    }

    sql1=date_data_query[self.loc]
    sql2=overall_query[self.loc]
    conn = mysql.connector.connect(
        host="covidsentimentanalysis.mysql.pythonanywhere-services.com",
        user="covidsentimentan",
        password="administrator",
        database = "covidsentimentan$tweets"
    )
    cursor = conn.cursor()
    cursor.execute(sql1)
    self.all_data = cursor.fetchall()

```

```

cursor2 = conn.cursor()
cursor2.execute(sql2)
self.temp_overall_data = cursor.fetchall()
self.overall_data=[]
self.overall_data.append(int(self.temp_overall_data[0][0]))
self.overall_data.append(int(self.temp_overall_data[0][1]))
self.overall_data.append(int(self.temp_overall_data[0][2]))
self.overall_data.append(int(self.temp_overall_data[0][3]))
self.overall_data.append(int(self.temp_overall_data[0][4]))
self.overall_data.append(int(self.temp_overall_data[0][5]))
print(self.overall_data)
#print(sum(self.temp_overall_data))
# j=0
# while(j<6):
#     sum=0
#     i=0
#     while(i<len(self.temp_overall_data)):
#         sum+=self.temp_overall_data[i][j]
#     self.overall_data.append(sum)
conn.close
# to convert row into convert

def getdata(self):
if len(self.all_data)>7:
    x=len(self.all_data)-7
    self.all_data = self.all_data[x:]
dates=[]
pos=[]
neg=[]
nut=[]
sad=[]
ang=[]
joy=[]
total_data=[dates,pos,neg,nut,sad,ang,joy]
for data in self.all_data:
    i=0
    while(i<7):
        total_data[i].append(data[i])
    i=i+1

```

It will fetch data from the last seven rows of a table, as our sites get update after 24hrs.

## EXPERIMENTAL SETUP AND RESULTS

These are the observations that were found after we analyzed tweets from 21-06-2020 to 17-07-2020. We analyzed approximately 43,000 tweets:

Positive Sentiment:6731

Negative Sentiment:13135

Neutral Sentiment:22617



## ADVANTAGES

The All over data section in web-app visualizes data according to dates.

And Major Cities section visualize data of major cities according to dates.

Our web-app also shows the latest tweets from “Ministry of Health” of India and “WHO”.

Web-app is user friendly and the design of the site is responsive i.e. it can be viewed on any device without facing any trouble.

## DISADVANTAGES

The main problem is that the NLP model is unable to detect a sarcastic tone in the text.

The drawback of our web-app is, it unable to recognize language other than English.

## CONCLUSIONS

In our paper, we focus on using Twitter, the most popular microblogging platform, our purpose of this paper is to introduce a new strategy to identify how the peoples are responding to the government's decision of lockdown. In this paper we have proposed identifies the sentiments of the tweets with the relevant #tags of COVID.

## REFERENCES

- I. G. Adda, J. Mariani, J. Lecomte, P. Paroubek, and M. Rajman. 1998. The GRACE French part-of-speech tagging evaluation task. In A. Rubio, N. Gallardo, R. Castro, and A. Tejada, editors, LREC, volume I, pages 433–441, Granada, May.
- II. Ethem Alpaydin. 2004. Introduction to Machine Learning (Adaptive Computation and Machine Learning). The MIT Press.
- III. Hayter Anthony J. 2007. Probability and Statistics for Engineer and Scientists. Duxbury, Belmont, CA, USA.
- IV. Alec Go, Lei Huang, and Richa Bhayani. 2009. Twitter sentiment analysis. Final Projects from CS224N for Spring 2008/2009 at The Stanford Natural Language Processing Group.
- V. D. Factiva, "Quick Study: Direct Correction Established Between Social Media Engagement and Strong Financial Performance," PR News, 2009.
- VI. A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welp, "Predicting elections with twitter: What 140 characters reveal about political sentiment," *Icwsn*, vol. 10, no. 1, pp. 178-185, 2010.
- VII. R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Information Sciences*, vol. 181, no. 6, pp. 1138-1152, 2011/03/15/ 2011.
- VIII. A. Giachanou and F. Crestani, "Like It or Not: A Survey of Twitter Sentiment Analysis Methods," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 1-41, 2016.
- IX. A. Abirami and V. Gayathri, "A survey on sentiment analysis methods and approach," in *Advanced Computing (ICoAC)*, 2016 Eighth International Conference on, 2017: IEEE, pp. 72-76.
- X. V. Kharde and P. Sonawane, "Sentiment analysis of twitter data: A survey of techniques," arXiv preprint arXiv:1601.06971, 2016.
- XI. Alexander Pak, Patrick Paroubek 2019 Twitter as a Corpus for Sentiment Analysis and Opinion Mining Université de Paris-Sud, Laboratoire LIMSI-CNRS, Bâtiment 508, F-91405 Orsay Cedex, France.
- XII. Abdullah Alsaedi, Mohammad Zubair Khan 2019, A Study on Sentiment Analysis Techniques of Twitter Data Department of Computer Science, College of Computer Science and Engineering Taibah University Madinah, KSA
- XIII. Link of our Web-app: <https://covidssentimentanalysis.pythonanywhere.com/>
- XIV. Source code: <https://github.com/SmartPracticeschool/SBSPS-Challenge-4672-Sentiment-Analysis-of-COVID-19-Tweets-Visualization-Dashboard>