**SENTIMENT ANALYSIS OF COVID-19 TWEETS- VISUALIZATION DASHBOARD**

**FINAL REPORT**

**TEAM NAME :**

**THE PY-SQUAD**

**PROJECT ID:**

**SPS\_PRO\_1742**

**TEAM MEMBERS :**

**PRABHU U ( 111717107027)**

**NIRMITHA S (111717107025)**

**KAVIYA S S (111717107017)**

**SRI RAJESWARI S (111717107049)**

**COLLEGE NAME : R.M.K. ENGINEERING COLLEGE CHENNAI, TAMILNADU.**

**INTRODUCTION**

**OVERVIEW:**

Sentiment analysis is basically analyzing people’s emotions ( positive or maybe negative) towards a matter, be it a movie or an online product by analysing their feedback or reviews. It mainly uses NLP (Natural language processing) . It is text analysis method based on texts/documents/paragraph and helps the product developer to develop the product meeting the customer needs. It is based on polarity or feelings

TRADITIONAL STEPS INVOLVED ARE :

* Break a sentence into various components.
* Neglecting the punctuationa and other stop words.
* Identify the word and assign a sentiment score.
* Calculate the overall score to determine the sentiment of the statement.

ALGORITHMS:

LOW-LEVEL: it includes simple techniques such grouping similar words, removing unnecessary parts and assigning scores. It includes techniques such as Tokenization, Lmammetization, Data Cleaning.

AUTOMATIC: Low -level analytics may become uneffective at times and that is where machine learning comes into picture..Large amount of text can be used to train the model using supervised or unsupervised techniques. it contains training and prediction process which converts text inputs and fed to machine learning and generates predicted tags.

HYBRID: It is an efficient method which combines both low-level text analytics and machine learning algorithms to enhance the classification.

Some machine learning techniques involve Naïve bytes(baeyes’s theorem), Linear regression, Support vector machines and Deep learning.

**PURPOSE**:

The purpose of Sentiment analysis is to analyze feedbacks in an accurate format. It is extremely useful in social media monitoring as it allows us to gain an overview of the wider public opinion in an accurate form.

**LITERATURE SURVEY**

**EXISTING PROBLEM:**

**Resource:**<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7152888/>

Corona Virus Disease or COVID 19 is a new virus disease that originated in 2019.The virus has now spread across the world and the almost all the countries are battling against this virus and are trying their best to curb the spread as much as possible. The World Health Organisation has declared it as a pandemic and is leaving no stone unturned to control the pandemic and is awaiting a vaccine to cure it .By the first week of March 2020, several countries like China, Italy, Spain, and Australia were fighting with the COVID19 pandemic by taking strict measures like nationwide lockdown or by cordoning off the areas that were suspected of having risks of community spread. Taking cues from the foreign counterparts, the government of India undertook an important decision of nationwide lockdown on March 25th for 21days from March 26th to April 14th, 2020.

India, with a population of 1.3 Billion people, was at a high risk of suffering from irreversible damage, and strict measures were expected to “flatten the curve.” The Prime Minister of India announced the lockdown , but it did not come as a surprise because Indians were actually given a feel of what it had in store through a one-day curfew named as “Janata Curfew” of 14 h on March 22nd from 7 A.M. to 9 P.M.. Thus, Indians were exposed to a lockdown situation partially, and this helped in preparing mentally for the nationwide lockdown, and the announcement did not come as a shocker to them.

Sentiment analysis of Indians 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 two prominent hashtags used namely: #IndiaLockdown and #IndiafightsCorona from March 25th to March 28th 2020. A total of 24,000 tweets were considered for the analysis. Analysis was done using the software R and a wordcloud was generated that depicts the sentiments of the tweets.Even though there was negativity, fear, disgust, and sadness about the lockdown, the positive sentiments stood out.

Overall, it can be seen that Indians have taken the fight against COVID19 positively and 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. Nevertheless, as of now, the lockdown response seems positive and indicates that India has succeeded in controlling the corona virus spread to a great extent.

**PROPOSED SOLUTION:**

We have developed a sentiment analysis model to understand the following:

• To know the people’s opinion towards the epidemic.

• To understand the people’s sentiment towards the government decisions.

We have done our project to meet the above requirements by the following ways:

The unstructured texts are classified by using NLP techniques and applying machine learning algorithms .

They are as follows:

➢ **Preprocessing**: The raw text are mined and Machine learning algorithms are applied to the text. The main objective of this step is to clean noise such as punctuation, special characters, numbers etc. Thus the following steps are:

* **Tokenisation** : This method I used to split strings into smaller parts called tokens. The splitting is done based on whitespace and punctuation.
* **Normalization** : This method group together the same meaning but different forms . Thus it includes 2 popular techniques called Lemmatization and Stemming
* **Lemmatization** : This process is used to normalize words with morphological analysis of words in text .
* **Stemming** : This process is to remove affixes from a word and it works with verb forms.
* **Remove noise** : This method removes the stop words(meaningless words eg. Is,the, a)

Removes punctuations and special characters

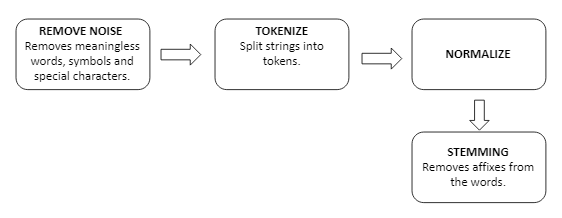
Removes hyperlinks and @username

* **Extraction/Vectorization of data**: This method is used to covert texts to numbers as Machine learning deals with numbers.
* **TF-IDF : Term frequency – Inverse document frequency which defines the importance of words Term Frequency :** This method depend on words that appears in document frequently Inverse Document frequency .This method depends on words that appears less frequently in the document because they are more important and informative. This basically reduces the common words usage .

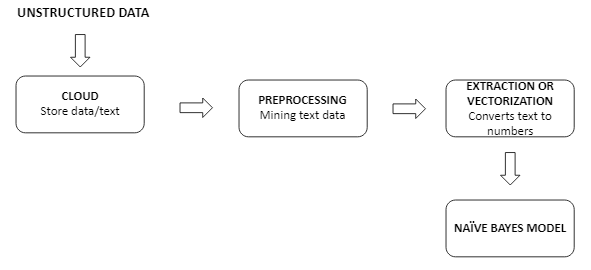
**Model : • Naive Bayes Model :** It is a classifier to classify text which is applied based on Bayes theorem Mainly uses Machine learning and data mining techniques to filter Irrelevant informations This Model performs better than other models and need less training data.It can handle both continous and discrete datas.

**THEORITICAL ANALYSIS**

**BLOCK DIAGRAM:**



PREPROCESSING



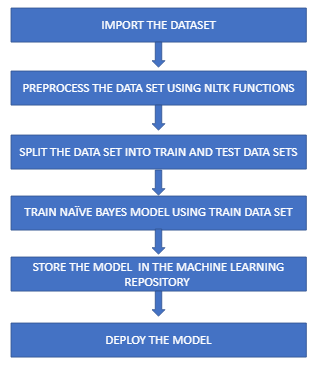
**HARDWARE/SOFTWARE REQUIREMENTS:**

* IBM CLOUD ACCOUNT
* IBM CLOUD SERVICES
* MACHINE LEARNING SERVICES
* JUPYTER NOTBOOK
* IBM EMBEDDED DASHBOARD

**EXPERIMENTAL INVESTIGATIONS**

During this project we learnt how to work on IBM Cloud and use it services. We learnt how to manage our project in IBM Watson, how to create a Visualization dashboard, and how to access machine learning services in IBM Cloud. We learnt about the usage of NLTK packages and in depth about sentiment analysis.We studied about Naive Bayes theorm. We learnt that the model accuracy depends on the amount of data set. Various models have different accuracy values.

**FLOWCHART**



**RESULT**

Our model has been successfully built. It stored in the Machine Learning repository and was sucessfully deployed. We also created the visualization dashboard using IBM Cognos to visualize our data.

**ADVANTAGES**

* Code can work on large datasets
* Usage of in-built python packages enables a lot of functions on the data set at an ease of access.
* Code is simple, effective and understandable.

**DISADVANTAGES**

* Detecting emoticons from the text can be difficult.
* Multilingual text cannot be predicted.
* Sarcastic text are difficult to predict.

## **APPLICATIONS**

## Social Media Monitoring (SMM) Understand social data like never before.

## People Analytics & Voice of Employee Reduce turnover, improve employee engagement, and raise productivity.

## Voice of Customer (VoC) & Customer Experience Management Transform mountains of unstructured customer feedback into useful data.

## Regulatory Compliance

## Solve regulatory compliance problems that involve complex text documents.

**CONCLUSION**

Our project made use of Natural Language Processing packages and sci-kit learn machine learning package to build the model. We downloaded the dataset from IEEE data port and hydrated fresh tweets using the access token. We used jupyter notebook and deployed the model in IBM Cloud. Finally we made a visualization using the IBM Cognos Dashboard.This will be able to analyze the sentiments of public towards the Government’s lockdown decisions. A positive attitude of the public towards the public is necessary for fighting this pandemic.

**FUTURE SCOPE**

* Through the use of Recurrent Natural Networks or Google Natural Language API, code can be implemented on multilingual texts.
* It can be used in all the industrial and institutional areas for analysing their customer’s reviews towards their products and services.
* Integrating with voice recognition services ,speech can also be analysed for its overall sentiment.
* Integrating with visual recognition , emoticons can also be detected that adds value to the sentiment.

**BIBILOGRAPHY**

# How To Perform Sentiment Analysis in Python 3 Using the Natural Language Toolkit (NLTK) By [Shaumik Daityari](https://www.digitalocean.com/community/users/sdaityari) <https://www.digitalocean.com/community/tutorials/how-to-perform-sentiment-analysis-in-python-3-using-the-natural-language-toolkit-nltk#:~:text=Sentiment%20analysis%20is%20a%20common,Python%2C%20to%20analyze%20textual%20data.>

## Python for NLP: Sentiment Analysis with Scikit-Learn By Usman Malik

<https://stackabuse.com/python-for-nlp-sentiment-analysis-with-scikit-learn/>

Building a Twitter Sentiment Analysis in Python By Gaurav Singhal

<https://www.pluralsight.com/guides/building-a-twitter-sentiment-analysis-in-python>

# Comprehensive Hands on Guide to Twitter Sentiment Analysis with dataset and code By Prateek Joshi <https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/>

Predicting Reddit News Sentiment with Naive Bayes and Other Text Classifiers By Brenden Martin and Nikos Koufus

<https://www.learndatasci.com/tutorials/predicting-reddit-news-sentiment-naive-bayes-text-classifiers/>

**APPENDIX**

**Source code**

1. **import pandas as pd**
2. **import numpy as np**
3. **import re**
4. **import string**
5. **import nltk**
6. **import matplotlib.pyplot as plt**
7. **from nltk.corpus import stopwords**
8. **from nltk.tokenize import word\_tokenize**
9. **from sklearn.feature\_extraction.text import TfidfVectorizer**
10. **from sklearn.model\_selection import train\_test\_split**
11. **from nltk.stem import PorterStemmer**
12. **from nltk.stem import WordNetLemmatizer**
13. **from sklearn.metrics import accuracy\_score**
14. **from sklearn.naive\_bayes import MultinomialNB**
15. **from sklearn.linear\_model import LogisticRegression**
16. **from sklearn.svm import SVC**
17. **nltk.download('stopwords')**
18. **nltk.download('punkt')**
19. **)**
20. **filtered\_words = [w for w in tweet\_tokens if not w in stop\_words]**
21. **ps = PorterStemmer()**
22. **stemmed\_words = [ps.stem(w) for w in filtered\_words]**
23. **return " ".join(filtered\_words)**
24. **stop\_words = set(stopwords.words('english'))**
26. **def preprocess(tweet):**
27. **tweet.lower()**
28. **tweet = re.sub('http[s]?://(?:[a-zA-Z]|[0-9]|[$-\_@.&+#]|[!\*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+','', tweet)**
29. **tweet = re.sub("(@[A-Za-z0-9\_]+)","", tweet)**
30. **tweet = tweet.translate(str.maketrans('', '', string.punctuation)**
31. **tweet\_tokens = word\_tokenize(tweet)**
32. **def result(score):**
33. **if score < 0:**
34. **return 'negative'**
35. **elif score == 0:**
36. **return 'neutral'**
37. **else:**
38. **return 'positive'**
39. **import types**
40. **import pandas as pd**
41. **from botocore.client import Config**
42. **import ibm\_boto3**
44. **def \_\_iter\_\_(self): return 0**
46. **# @hidden\_cell**
47. **# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.**
48. **# You might want to remove those credentials before you share the notebook.**
49. **client\_154336f42d194f25bb030b76e7f109fd = ibm\_boto3.client(service\_name='s3',**
50. **ibm\_api\_key\_id='ZY5bk4X9TizH7sKrJ7yNyKvSv9iWIV7ei78dWwXuCj9S',**
51. **ibm\_auth\_endpoint="https://iam.cloud.ibm.com/oidc/token",**
52. **config=Config(signature\_version='oauth'),**
53. **endpoint\_url='https://s3.eu-geo.objectstorage.service.networklayer.com')**
55. **body = client\_154336f42d194f25bb030b76e7f109fd.get\_object(Bucket='sentimentanalysisofcovid19tweets-donotdelete-pr-x01mtp2wbei3mi',Key='Ready corona tweets (2).csv')['Body']**
56. **# add missing \_\_iter\_\_ method, so pandas accepts body as file-like object**
57. **if not hasattr(body, "\_\_iter\_\_"): body.\_\_iter\_\_ = types.MethodType( \_\_iter\_\_, body )**
58. **tweets= pd.read\_csv(body)**
59. **index = tweets.index**
60. **columns = tweets.columns**
61. **values = tweets.values**
62. **tweets\_new=tweets[['tweet','senti\_score']]**
63. **tweets\_new['Result']=tweets\_new['senti\_score'].apply(result)**
64. **tweets\_newvectorizer = TfidfVectorizer(sublinear\_tf=True)**
65. **x = vectorizer.fit\_transform(tweets\_new.tweet)**
66. **y= tweets\_new.Result**
67. **x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.20,random \_state=0)**
68. **NB\_model = MultinomialNB()**
69. **NB\_model.fit(x\_train, y\_train)**
70. **y\_predict= NB\_model.predict(x\_test)**

# 