**Coronavirus Tweet Sentiment Analysis**

**Suraj Kumar**

**Shreya Ranjan**

**Data Science Trainee**

**AlmaBetter, Bangalore**

**Introduction:**

The virus (more precisely, a coronavirus) known as 2019 Novel Coronavirus (2019-nCoV) has been determined to be the origin of an outbreak of respiratory illnesses that was originally discovered in Wuhan, China. On March 11, 2020, the World Health Organization declared the virus to be pandemic. While some of us tried to resist this new strain, known as Covid-19, many of us did.

Viruses that originated in birds, pigs, bats, and other animals and transformed to become deadly to people have been linked to a number of infectious disease epidemics in the past. Research is ongoing, and further investigation may shed light on how and why the coronavirus changed over time to produce pandemic sickness.

Sentiment analysis is the technique of computationally identifying and classifying opinions stated in a piece of text, particularly to ascertain if the writer has a positive, negative, or neutral attitude toward a given topic. We will forecast the sentiments of COVID-19 tweets in this research. To implement this project, I'll use the data gathered from Tweeter and the python environment.

1. **Problem Statement:**

Building a classification model to predict the sentiment of Covid-19 tweets is the task at hand. The tweets have been manually tagged after being downloaded from Twitter. Information like Location, Tweet At, Original Tweet, and Sentiment have been provided.

1. **Data Description:**

* **Data set name:** Coronavirus tweets.csv
* **Shape :**
* Rows: 41157
* Columns: 6
* **Columns:** UserName, ScreenName, Location, TweetAt, OriginalTweet, Sentiment.
* **Username:** Twitter handle username
* **ScreenName:** The name that users choose to identify themselves on the network.
* **Location:** Place (Country) from where the tweets came from
* **OriginalTweet:** The content of the posted tweet
* **TweetAt:** Date in which the tweet has been posted
* **Origment:** Behaviour at the tweets of the people. There are five types of sentiments- Extremely Negative, Negative, Neutral, Positive, and Extremely Positive.

## **Steps involved :**

## **Performing EDA (exploratory data analysis) : -**

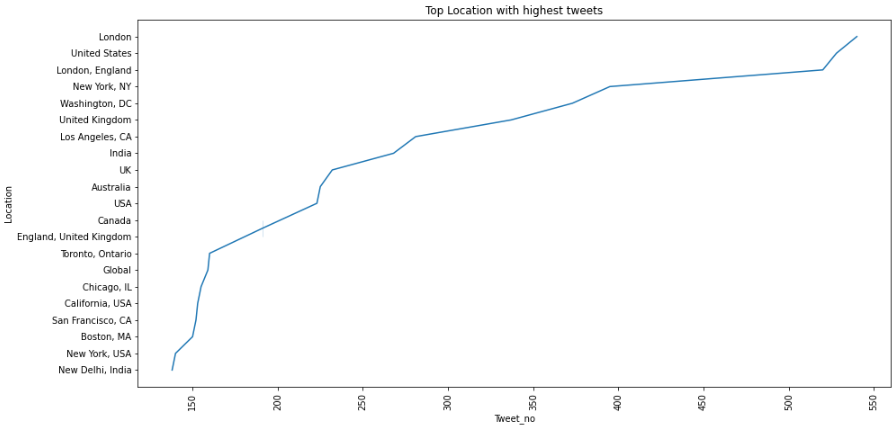
* Analyzing the data's head and tail to gain understanding of the information provided.
* Looking for null values, we don't need to delete the null values from the Location column Because we need to categorise the tweet based on the text in the OriginalTweet column.
* Classifying the 5 labels in the Sentiment column into 3 categories, positive, negative, and neutral.
* converting the TweetAt column's string format using a function to date-time format.
* By using the given text's stemming function. Stemming is a process that reduces a term to its root word or stem so that words of a similar kind are grouped under a single stem. The words "cared" and "bumped," for instance, will be stemmed as "care" and "bump."

## **Conclusions drawn from the data: -**

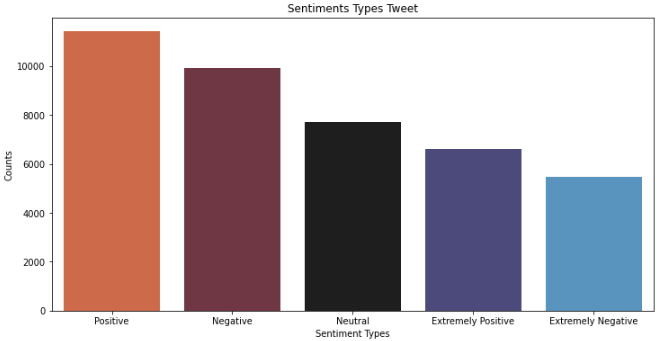
Plotting the needed graphs that provide the necessary information about our data, such as:

* Getting a plot on Top 20 Location with highest tweets.
* Plotting Seaborn Barplot Count of tweets as per sentiment.
* Plotting graph for the total number of tweets for the new 3 subcategories of the target variable.
* Getting a plot on Top dates with maximum number of tweets.

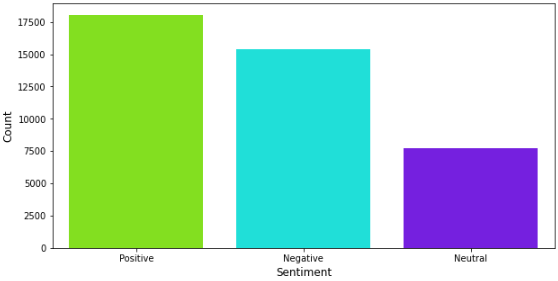
Getting a plot on Top 20 Location with highest tweets



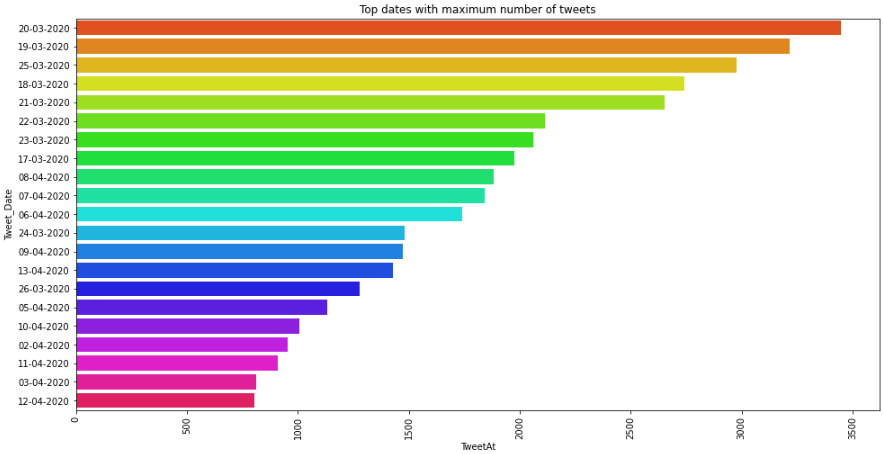
Plotting Seaborn Barplot Count of tweets as per sentiment.



Plotting graph for the total number of tweets for the new 3 subcategories of the target variable.



Getting a plot on Top dates with maximum number of tweets



## **Train And Test the Model:-**

* Vectorizing the text in OriginalTweet using countvectorizer. A countvectorizer is used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text.
* The dependent and independent variables are assigned.
* Splitting the model into train and test sets.
* Fitting logistic regression on train set.
* Obtaining the model's predicted variables.

## **Evaluating our model's metrics:-**

* Obtaining the confusion matrix for the train and test sets.
* Evaluating different scores such as accuracy score, precision score,

recall score, f1 score and support.

* Obtaining the classification report for our model.

## **Building Classification Models:**

There are five types of sentiments so we have to train our model so that it can give us the correct label for the test dataset.

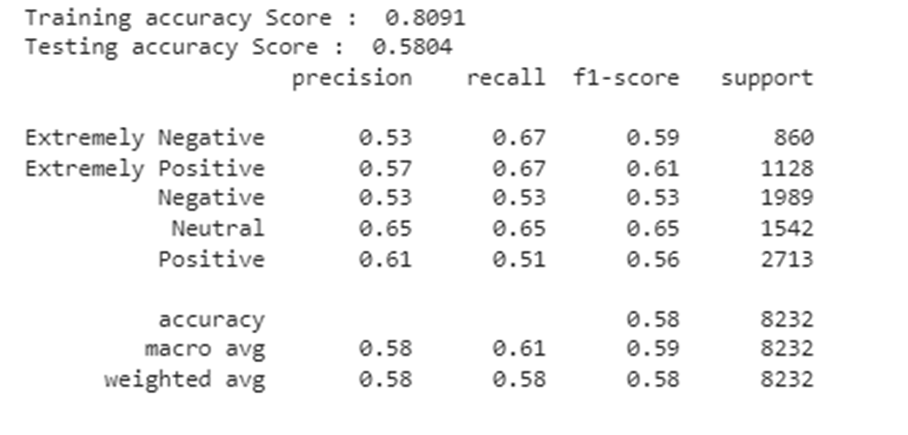
**We will implement 4 models:**

* Logistic Regression
* Random Forest Classifier
* Naive Bayes Classifier
* Support Vector Machine(SVM)

1. **Logistic Regression:-**

Logistic regression is a Machine Learning classification algorithm that is used to

predict the probability of a categorical dependent variable. The dependent variable in logistic regression is a binary variable with data coded as 1 (yes, success, etc.) or 0 (no) (no, failure, etc.). In other words, P(Y=1) is predicted by the logistic regression model as a function of X.

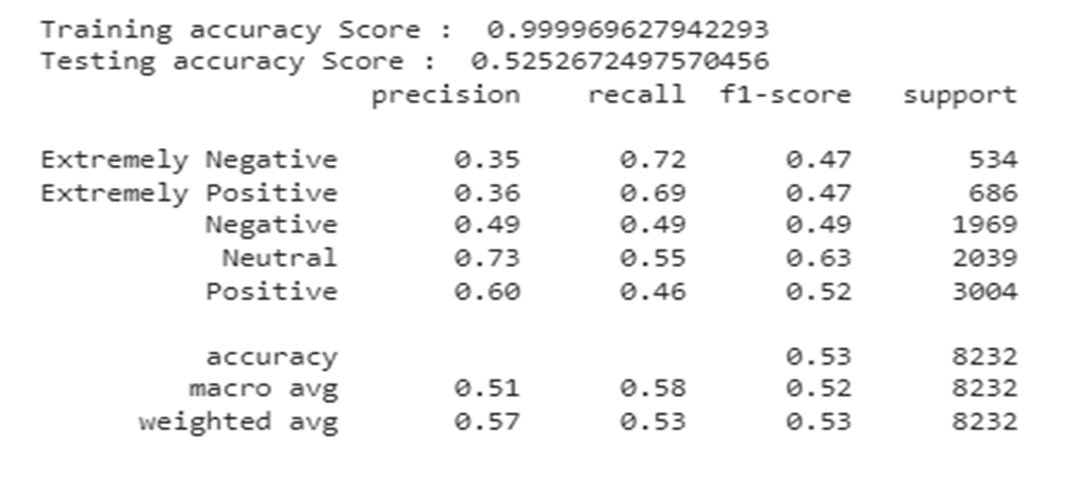


Our model comes under multinomial logistic regression as there are 3 categories positive , negative , neutral.

Evaluating different scores such as accuracy score(training & testing), precision score, recall score, f1 score and support.

1. **Random Forest Classifier:-**

Random Forest is a popular machine learning algorithm, It is a component of the supervised learning technique. Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

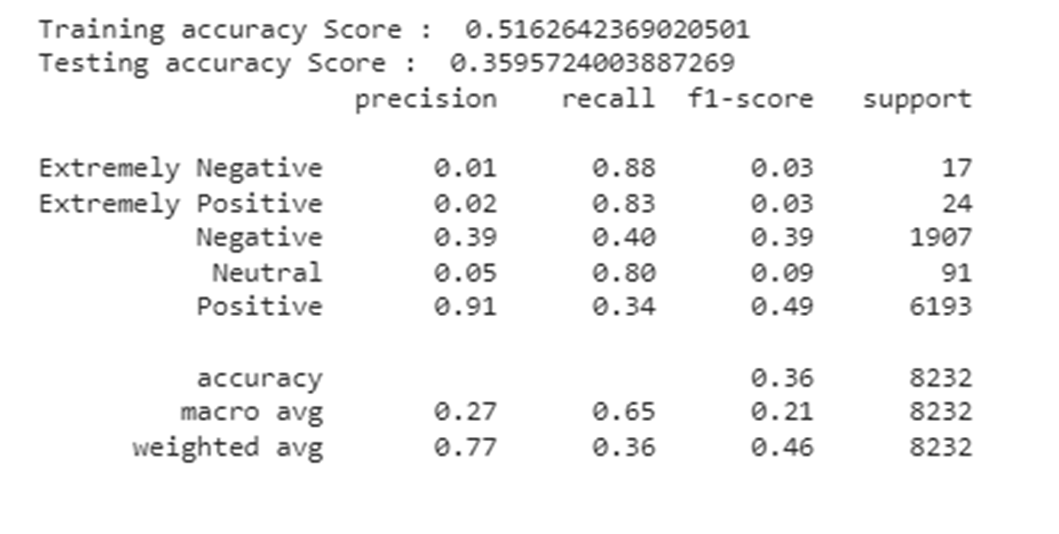


Evaluating different scores such as accuracy score(training & testing), precision score, recall score, f1 score and support.

1. **Naive bayes classifier:-**

Natural Language Processing uses the Multinomial Naive Bayes algorithm as a probabilistic learning technique most frequently (NLP). The method, which predicts the tag of a text such as an email or newspaper article, is based on the Bayes theorem. For a given sample, it determines the probabilities of each tag, and then outputs the tag with the highest probability.

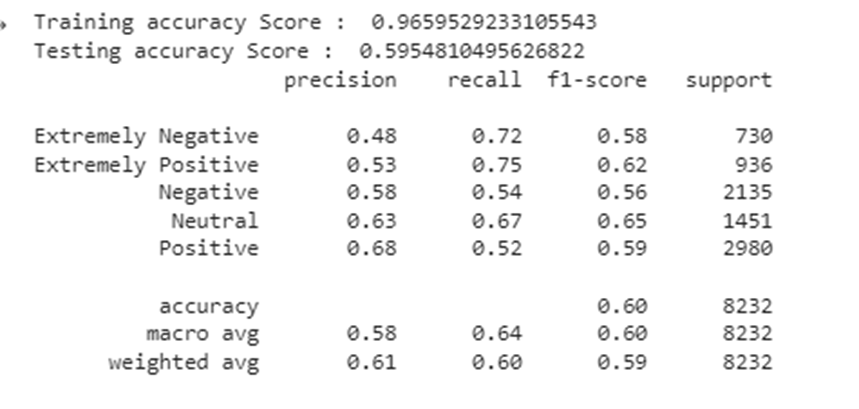
The Naive Bayes classifier is a collection of many methods, all of which are based on the concept that each feature being classified is independent of every other feature. The existence or absence of one feature has no impact on the other feature's existence or absence.



Evaluating different scores such as accuracy score(training & testing), precision score, recall score, f1 score and support.

1. **Support Vector Classifier (SVC):-**

The supervised machine learning technique known as SVC, or Support Vector Classifier, is frequently used for classification problems. SVC separates the data into two classes by mapping the data points to a high-dimensional space and then locating the best hyperplane.

****

Support Vector Classifier has performed slightly better than the Logistic regression and got the highest test accuracy score around 60%.

Multinomial Naive Bayes performed the worst with a test accuracy score of just 0.35.

## **Conclusion:**

* We are finally at the conclusion of our project!
* We performed EDA on the dataset right away and also cleaned the data to meet our requirements. Then, using the provided data, we were able to make pertinent deductions, and we trained our model using logistic regression.
* The "Sentiment" section informed me that most individuals had good feelings about a variety of topics, demonstrating their optimism in the face of pandemics. Extremely pessimistic ideas regarding Covid-19 are common.
* Support Vector Classifier has performed slightly better than the Logistic regression and got the highest test accuracy score around 60%.
* Multinomial Naive Bayes performed the worst with a test accuracy score of just 0.35.

## **References**:

* Analytics Vidya
* My Great Learning