

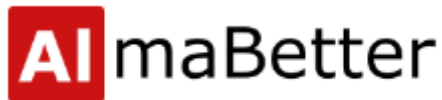
Capstone Project

On

Coronavirus Tweet Sentiment Analysis

By

Premanand Gaikwad



CONTENT

Following is the Standard Operating Procedure to tackle the Sentiment Analysis kind of project. We will be going through this procedure to predict what we supposed to predict!

- 1. Problem Statement**
- 2. Data Summary**
- 3. Exploratory Data Analysis (EDA)**
- 4. Text Pre-processing**
- 5. Classification Analysis**
- 6. Models Performance Metrics**
- 7. Conclusion**

Problem Statement

Sentiment Analysis is the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic is Positive, Negative, or Neutral. The given challenge is to build a classification model to predict the sentiment of Covid-19 tweets. The tweets have been pulled from Twitter and manual tagging has been done.

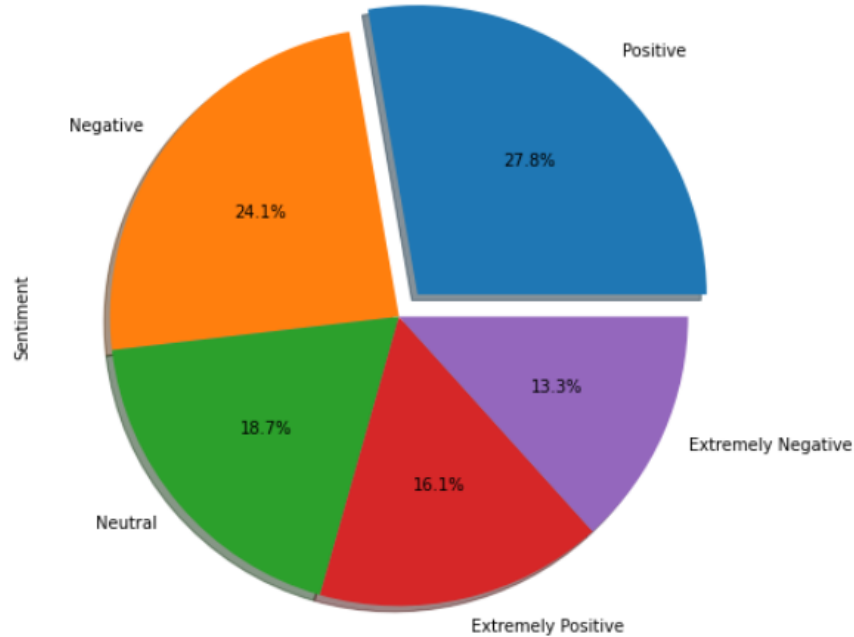
Data Summary

- The original dataset has 6 columns and 41157 rows. In order to analyze various sentiments, from this 6 feature 2 features are unusable so will ignore them
 1. Location = location (country) from where tweet is posted
 2. Tweet At = Date on which tweet is posted
 3. Original Tweet = Context of tweet
 4. Label = Type of sentiments

- We require just two columns named Original Tweet and Sentiment. There are five types of sentiments- Extremely Negative, Negative, Neutral, Positive, and Extremely Positive.

Exploratory Data Analysis (EDA)

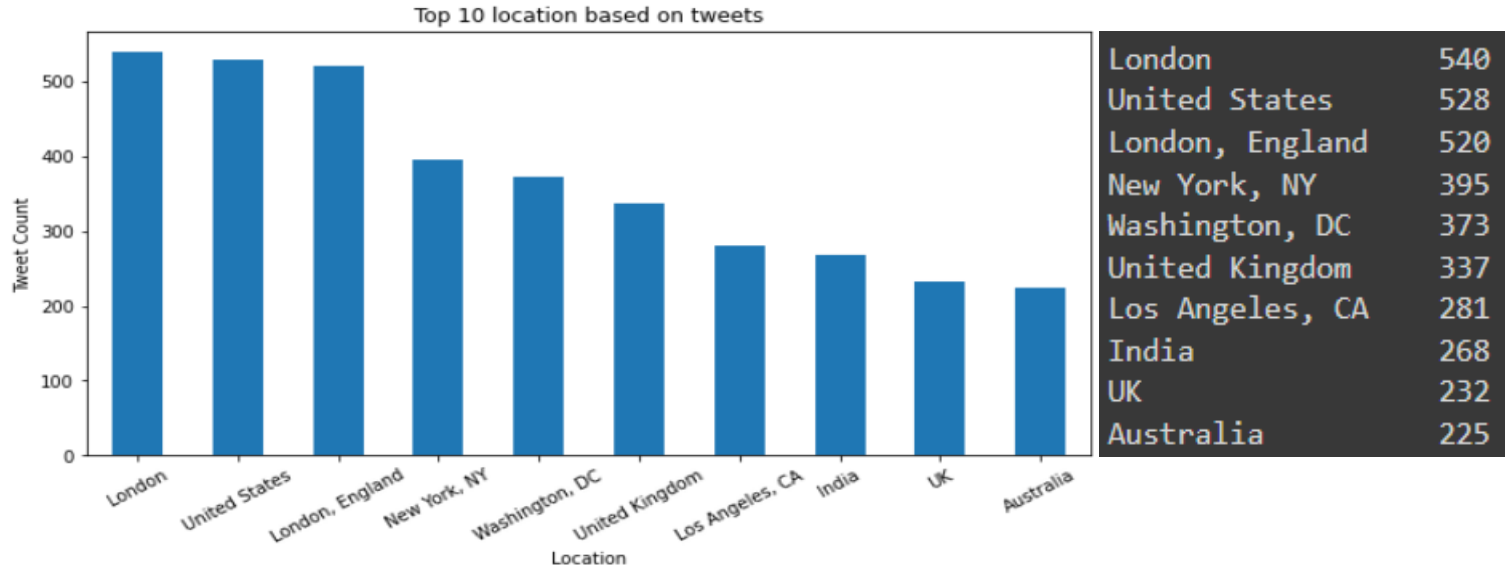
Percentage wise sentiments



When we try to explore the 'Sentiment' pie chart, we came to know that:

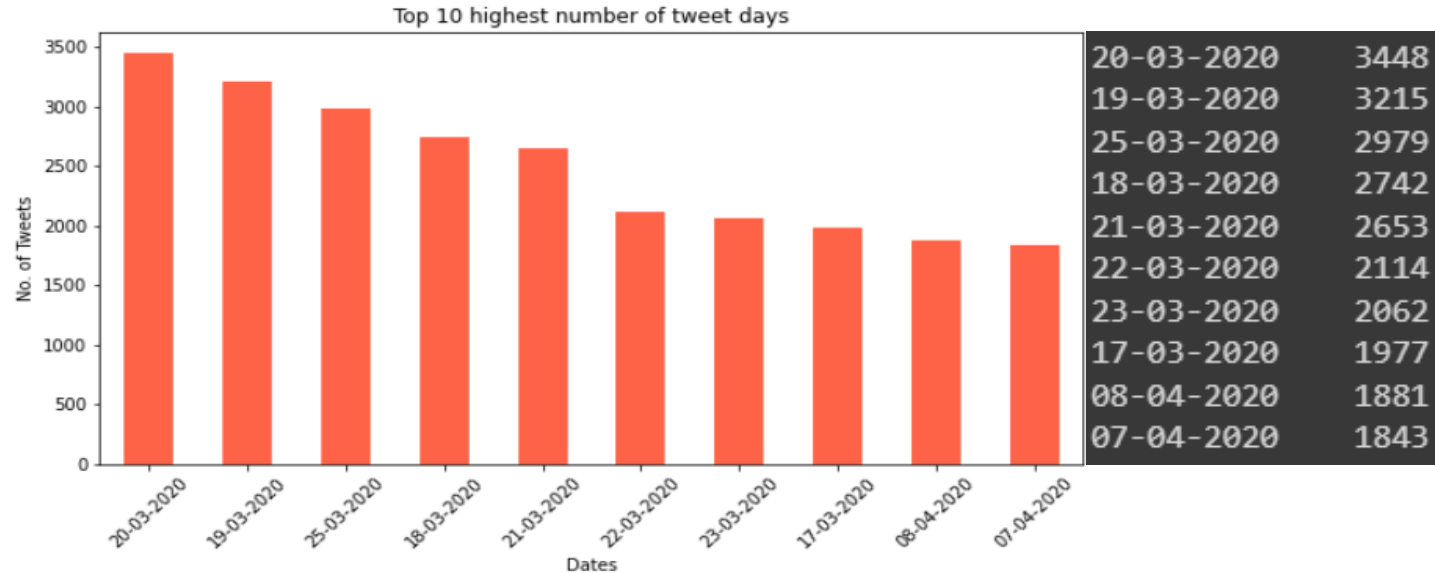
- ✓ Most of the peoples about 27.8% are having positive sentiments about various issues shows us their optimism during pandemic times.
- ✓ Very few people about 13.3% are having extremely negatives thoughts about Covid-19.

Top 10 location based on the no. of tweet



There are some null values in the location column but we don't need to deal with them as I am just going to use two columns i.e. "Sentiment" and "Original Tweet". Maximum tweets came from London(11.7%) location as evident from the above graph.

Top 10 highest number of tweet days

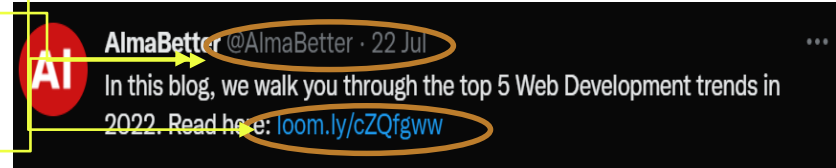


- ✓ From above graph it is seen that maximum number of tweets are posted in march and in march highest no. of tweet post are on the date 20th

Text Pre-Processing

Text pre-processing of the text data is an essential step as it makes the raw text ready for mining and making it suitable for a machine learning model. The objective of this step is to clean noise those are less relevant to find the sentiment of tweets such as :

- ✓ Url links (HTTPS: / HTTP:)
- ✓ Username/tweeter handle (@Xyz)
- ✓ Punctuation (.,?,” etc.),
- ✓ Special characters (@,%,&,\$, etc.),
- ✓ Numbers (1,2,3, etc.)



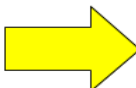
Terms which don't carry much weightage in context to the text are:

- **Stop words** are those words in natural language that have very little meaning, such as “is”, “an”, “the”, etc. To remove stop words from a sentence, divide your text into words and then remove the word if it exists in the list of stop words provided by NLTK

- **Stemming** is a rule-based process of stripping the suffixes (“ing”, “ly”, “es”, “ed”, “s” etc) from a word. It normalize the word. For example — “play”, “player”, “played”, “plays” and “playing” are the different variations of the word — “play”.
- **Tokenization** is essentially splitting a phrase, sentence, paragraph, or an entire text document into smaller units, such as individual words or terms. Each of these smaller units are called tokens.
- **Encode the Sentiments** to produce binary integers of 0 and 1 to encode our categorical features, because categorical features that are in string format cannot be understood by the machine and needs to be converted to numerical format.
 - 0 = Neutral sentiments
 - 1 = Positive and extremely positive sentiments
 - 1 = Negative and extremely negative sentiments

Vectorization

TFIDF, short for term frequency-inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. Vectorization is a step in feature extraction, the idea is to get some distinct features out of the text for the model to train on, by converting text to numerical vectors



Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1

Classification Analysis

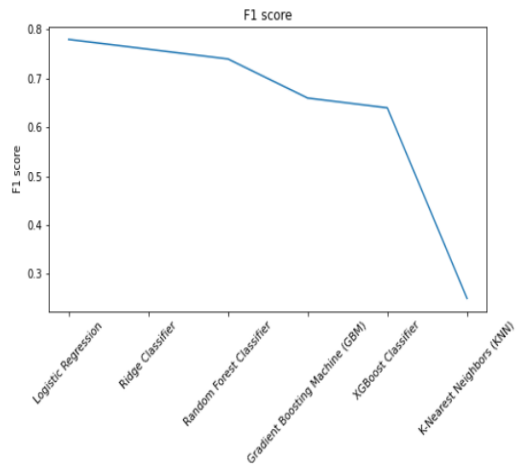
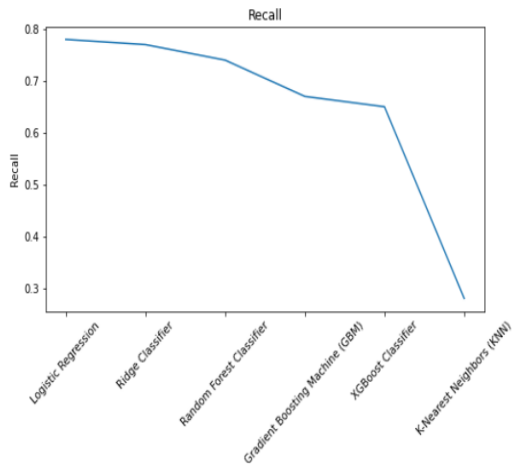
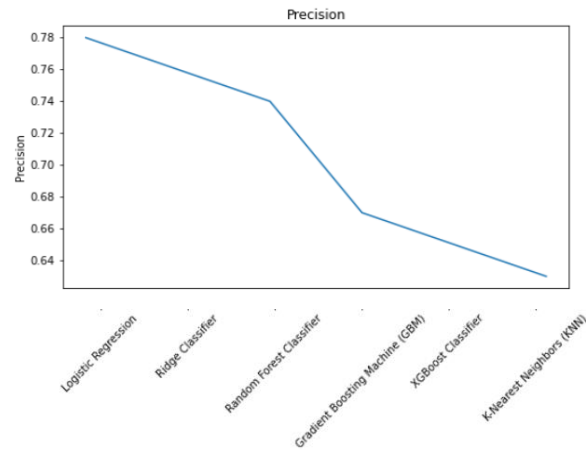
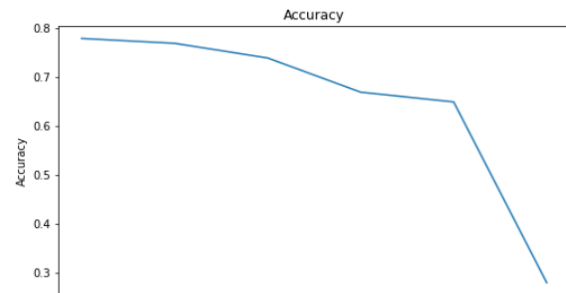
Building Classification Models

The given problem is Ordinal Multiclass classification. We had five types of sentiments and we converted them into three type, We have trained our models on different classification models are:

1. Logistic Regression
2. Ridge Classifier
3. K-Nearest Neighbors (KNN)
4. XGBoost
5. Gradient Boosting Classifier (GBC)
6. Random Forest

Models Performance Metrics

	Model_Name	Accuracy	Precision	Recall	F1 score
0	Logistic Regression	0.78	0.78	0.78	0.78
1	Ridge Classifier	0.77	0.76	0.77	0.76
2	Random Forest Classifier	0.74	0.74	0.74	0.74
3	Gradient Boosting Machine (GBM)	0.67	0.67	0.67	0.66
4	XGBoost Classifier	0.65	0.65	0.65	0.64
5	K-Nearest Neighbors (KNN)	0.28	0.63	0.28	0.25



Conclusion

- ✓ K-Nearest Neighbors (KNN) doesn't work well with a large dataset and with a high number of dimensions. It didn't classify the sentiments efficiently and ended up affecting the evaluation metrics and giving worse results than all the other implemented models.
- ✓ The Ridge classifier decreases the complexity of a model, and in the Random Forest classifier, the large number of trees makes the algorithm too slow. Both these models gave a moderate result of about 0.76 and 0.74 F1-score, respectively. The Gradient Boosting classifier (GBM) and XGBoost classifier gave almost identical results of 0.66 and 0.64 F1-score, respectively.
- ✓ Logistic regression gives the highest result of about 0.78 F1-score of all the implemented models, so I can use the logistic regression model for further classification.
- ✓ As I have seen above, while selecting a model, it should have good explainability and less complexity. As per the result, I have all three models with higher accuracy and less error, which is good explainable so that our final model can be the logistic regression.

Thank You