Capstone Project: 03

Sentiment Analysis :

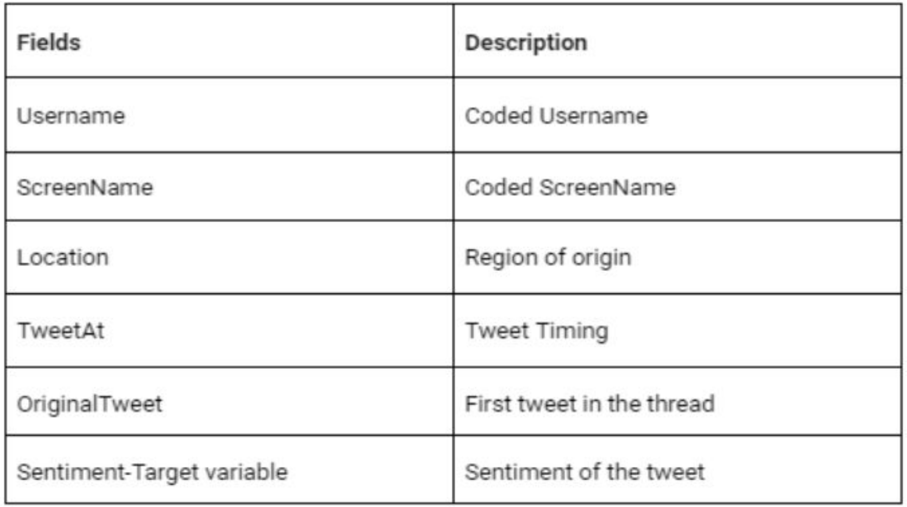
Predicting Sentiment of COVID-19 tweets

By

Pankaj Beldar

Problem Description

This challenge asks you 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 then. The names and usernames have been given codes to avoid any privacy concerns. You are given the following information:



Fields	Description
Username	Coded Username
ScreenName	Coded ScreenName
Location	Region of origin
TweetAt	Tweet Timing
OriginalTweet	First tweet in the thread
Sentiment-Target variable	Sentiment of the tweet

Sentiment Analysis

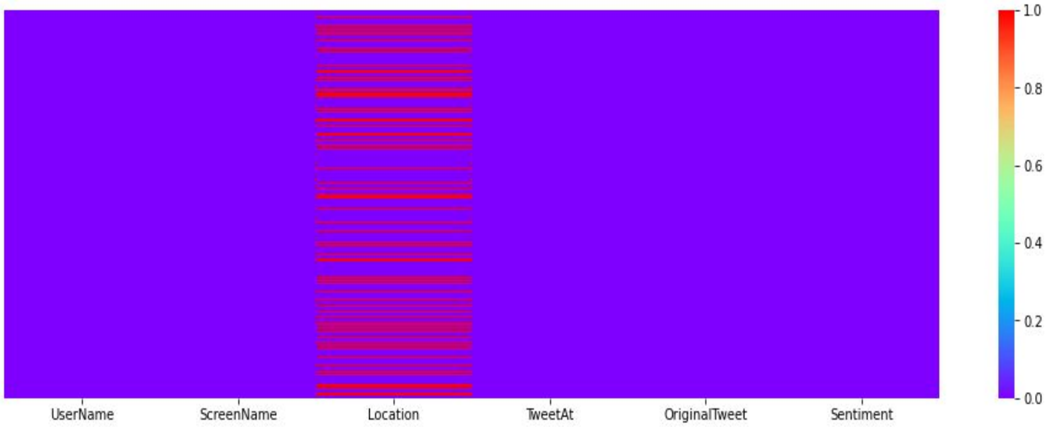
The Natural Language API breaks up text into its constituent words and punctuation (called tokens) and then provides information on each part. You can use the API to perform the following tasks on a chunk of text:

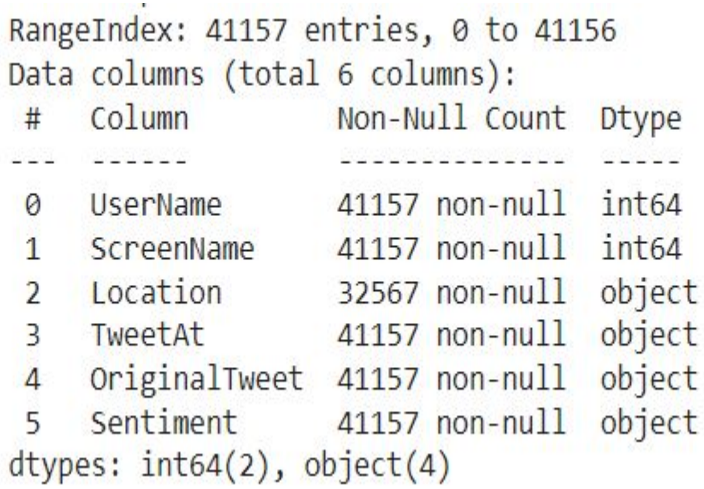
- **Syntax analysis-** identify parts of speech
- **Entity recognition-** label entities by type (person, location, event, etc.)
- **Sentiment analysis-** get the overall sentiment of a block of text
- **Content classification-** classify documents into predefined categories Upon reviewing the Natural Language API docs, I became most interested in sentiment analysis. Let's delve deeper into how this feature works

Sentiment analysis, also referred to as opinion mining, is **an approach to natural language processing (NLP) that identifies the emotional tone behind a body of text**.

Data Overview

Dataset consists of **41157 entries** and **6 attributes.** There are no Missing values in all columns except 'Location'. The attribute **'Location' has 8590 missing values.**



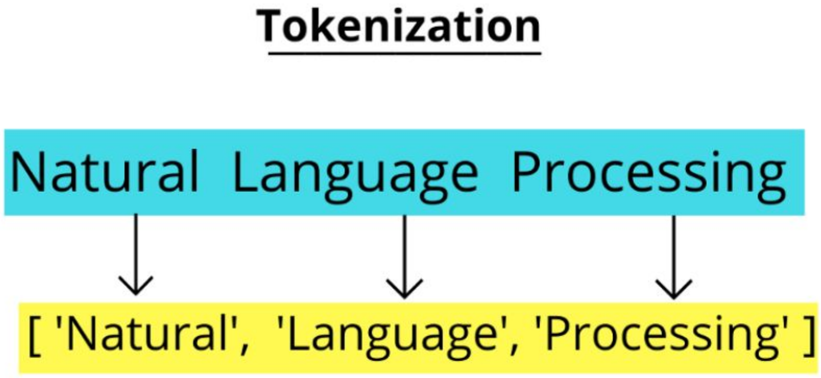


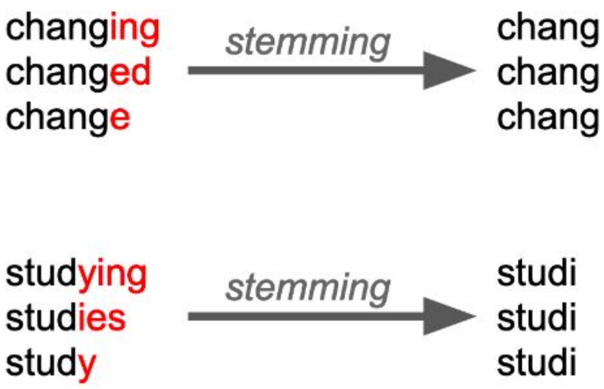
```
RangeIndex: 41157 entries, 0 to 41156
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   UserName       41157 non-null  int64 
 1   ScreenName     41157 non-null  int64 
 2   Location       32567 non-null  object
 3   TweetAt        41157 non-null  object
 4   OriginalTweet  41157 non-null  object
 5   Sentiment      41157 non-null  object
dtypes: int64(2), object(4)
```

Data Pre Processing

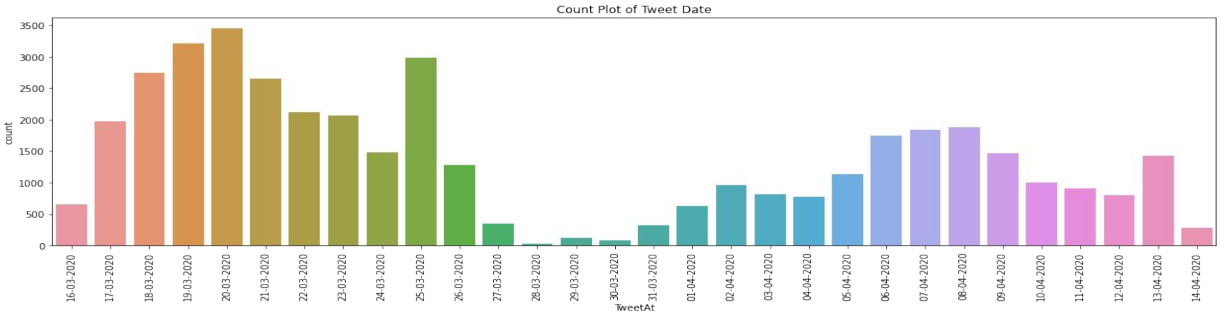
In data cleaning we have done following things-

1. Removed @users from tweets
2. Removed Punctuations
3. Removed Stopwords
4. Removed HTML tags
5. Removed URLs
6. Removed Short words
7. Removed Emojis
8. Convert text to Lower Case
9. Stemming
10. Tokenizing
11. Spelling Correction


Tokenization
Natural Language Processing
['Natural', 'Language', 'Processing']

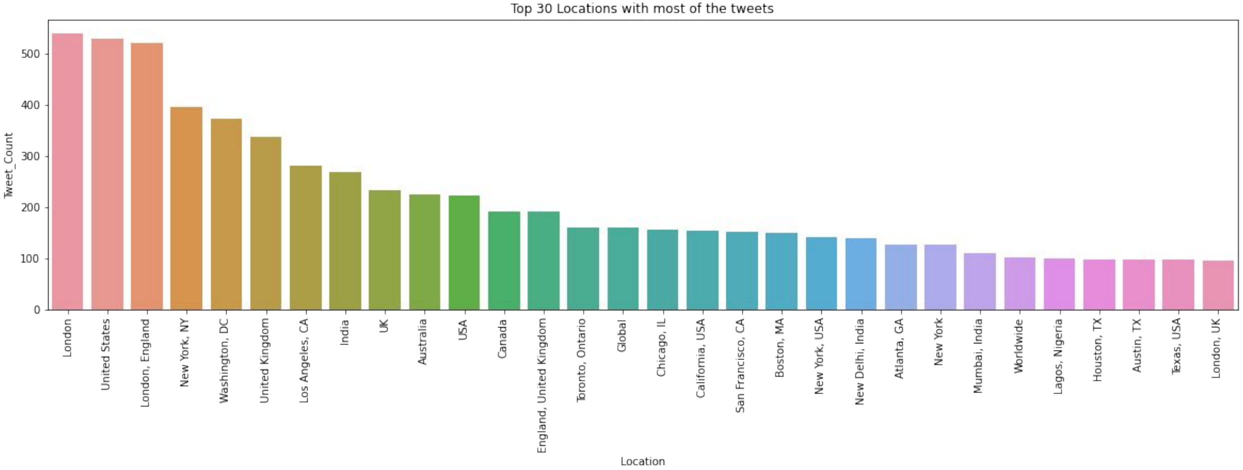


Exploratory Data Analysis



How many Tweets per day for the given time period?

We can see that maximum tweets were done on 20 march 2020, when the first lockdown was declared. People were more active on Twitter in the month of March because it was the early stage of Corona virus Pandemics and people wanted to know more about this disease.

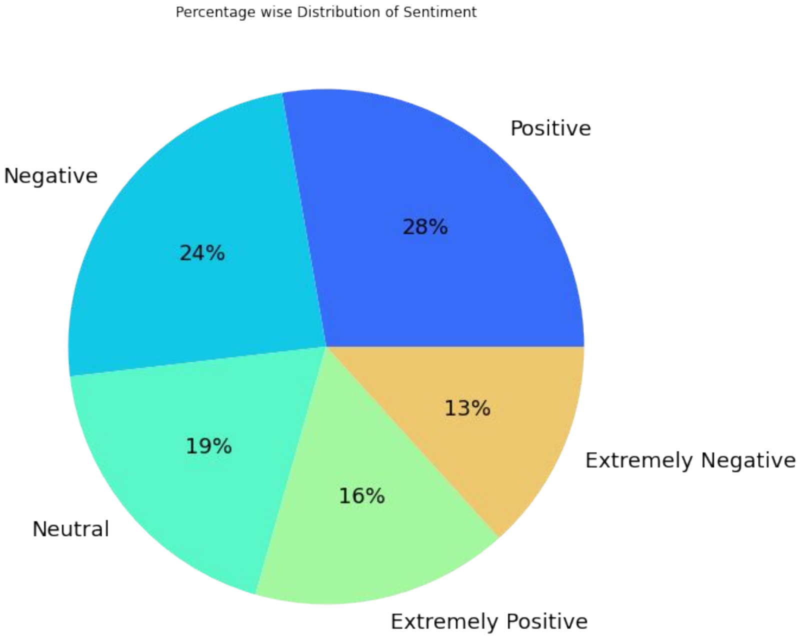


Which are the Top 30 Locations from where maximum tweets were done?

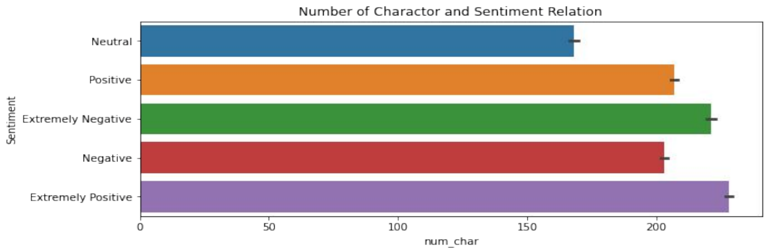
We can see London, United States, New York, Washington DC, United Kingdom ,India, Australia, USA are the top locations as far as count of tweets are concerned.

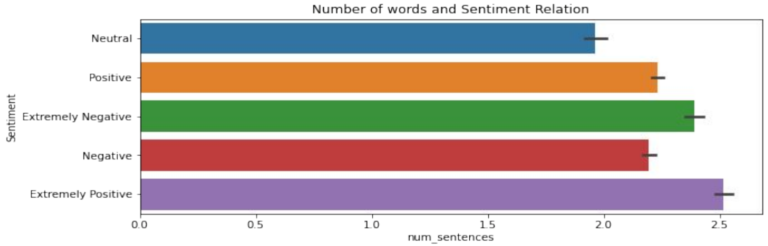
Exploratory Data Analysis

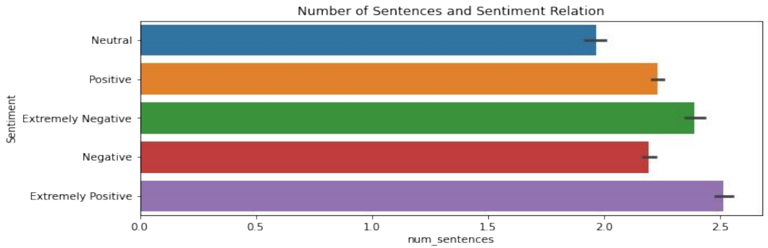
What are the Sentiment types and its Distribution?


Percentage wise Distribution of Sentiment

What is the relationship between the number of words, characters, sentences and different types of Sentiments?

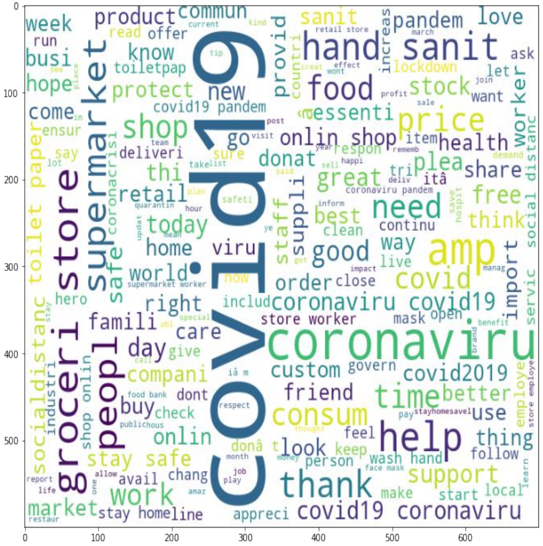

Number of Charactor and Sentiment Relation


Number of words and Sentiment Relation

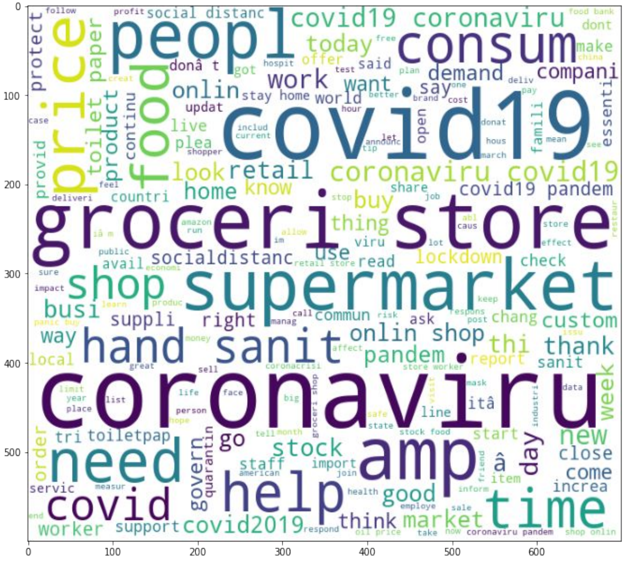

Number of Sentences and Sentiment Relation

Exploratory Data Analysis

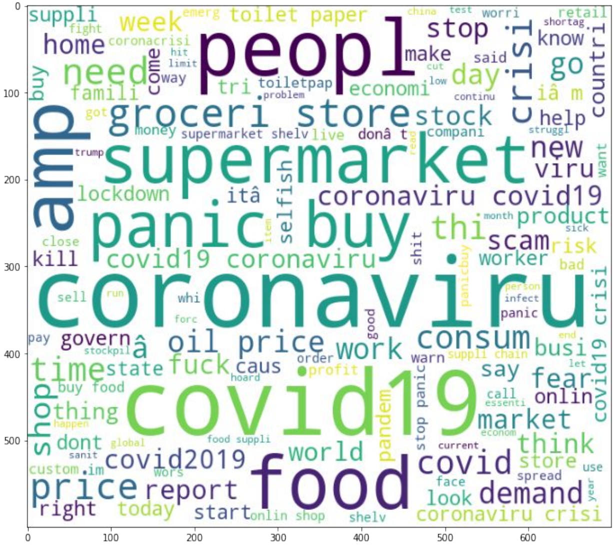
What are the most frequent Extremely Positive Sentiment Words?



What are the most frequent Positive Sentiment Words?

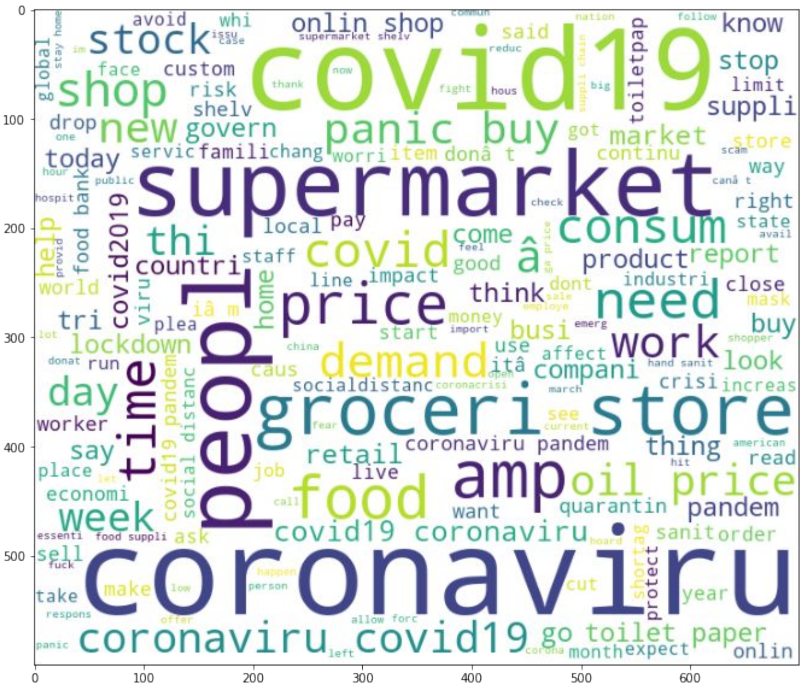


What are the most frequent Extremely Negative Sentiment Words?

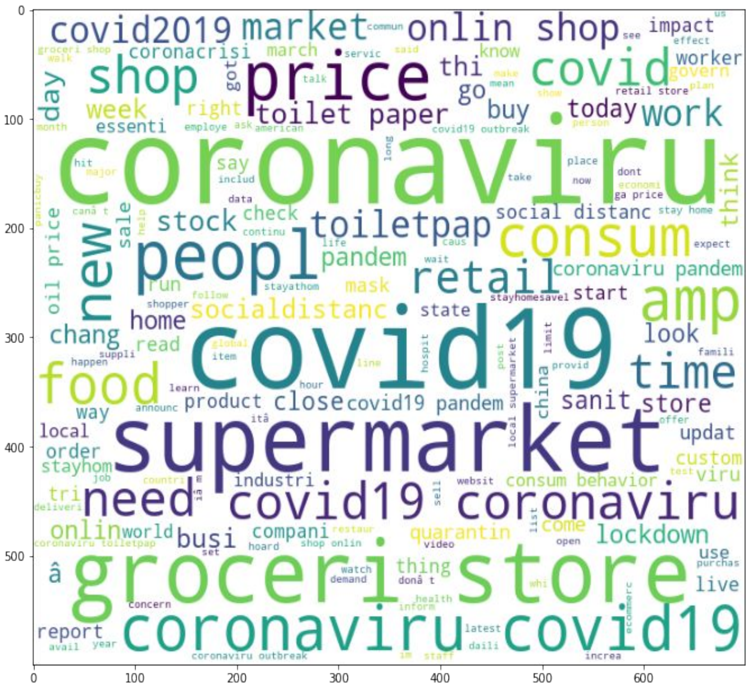


Exploratory Data Analysis

What are the most frequent Negative Sentiment Words?

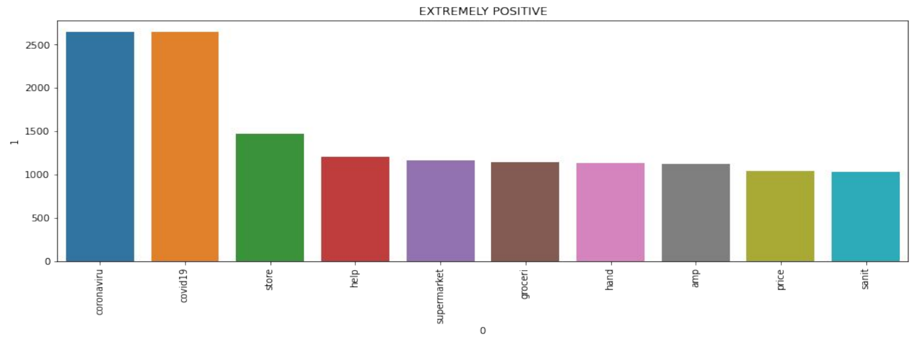


What are the most frequent Neutral Sentiment Words?

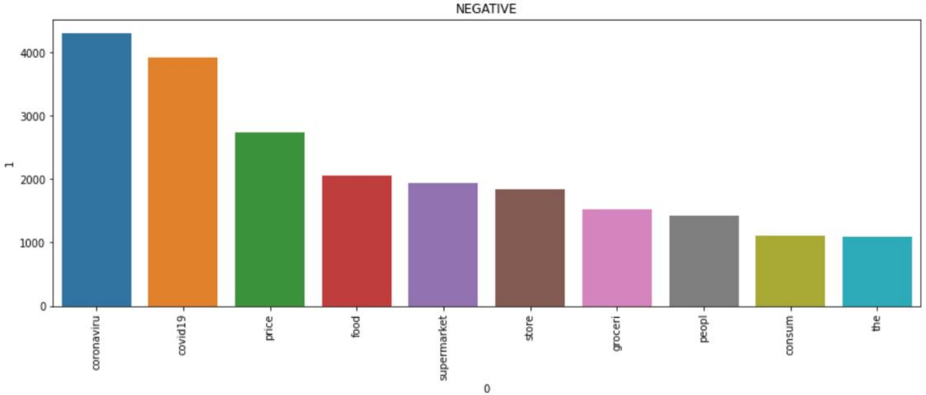


Exploratory Data Analysis

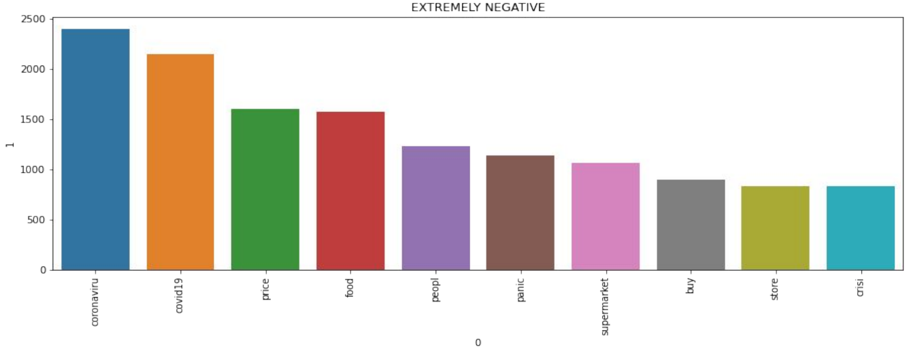
Finding out top 10 most frequent words based on sentiment.



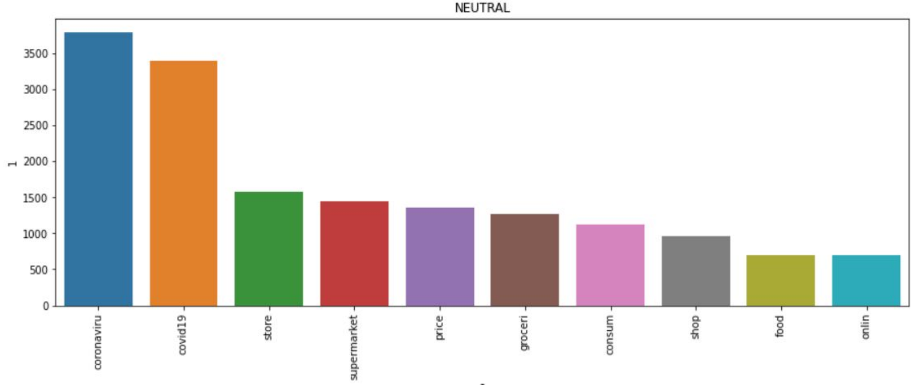
EXTREMELY POSITIVE



NEGATIVE



EXTREMELY NEGATIVE



NEUTRAL

Classification Model Development

- **Train Test Split (80-20)**

  train shape : (32925, 5)
  test shape : (8232, 5)

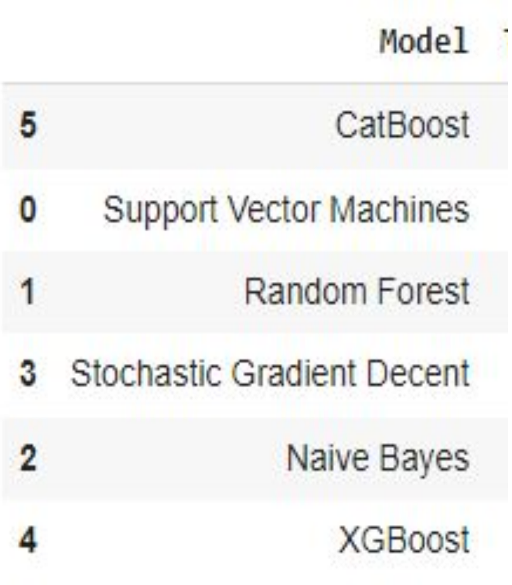
- **Feature Extraction (Vectorization)**

Word Embeddings or Word vectorization is **a methodology in NLP to map words or phrases from vocabulary to a corresponding vector of real numbers which used to find word predictions, word similarities/semantics**.
1.tfidfvectorizer
2.countvectorizer
3.n-grams

- **Feature selection**

1. As we want the sentiment analysis of the tweet, hence **'UserName', 'ScreenName', 'Location', 'TweetAt'** are **not playing an important role** for the determination of sentiment of the tweet.
2. Hence we select important features as '**Sentiment', 'num_char', 'num_words', 'num_sentences', 'Tweet'** for our analysis.

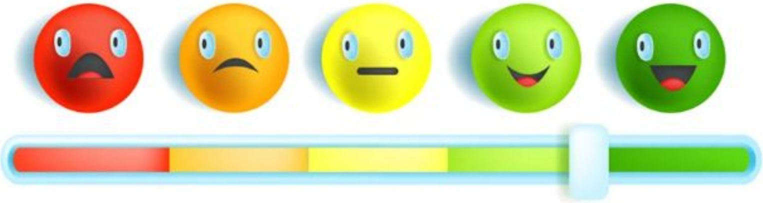
Multiclass Classification



	Model
5	CatBoost
0	Support Vector Machines
1	Random Forest
3	Stochastic Gradient Decent
2	Naive Bayes
4	XGBoost

Target Classes

1. Extremely Positive
2. Positive
3. Neutral
4. Extremely Negative
5. Negative



Multiclass Classification

➔ Naive Bayes classifier for multinomial models

The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts.

training accuracy Score : 0.64680

Validation accuracy Score : 0.48700

➔ SGDClassifier

The class SGDClassifier implements a plain stochastic gradient descent learning routine which supports different loss functions and penalties for classification. Below is the decision boundary of a SGDClassifier trained with the hinge loss, equivalent to a linear SVM.

Training accuracy Score : 0.77479

Validation accuracy Score : 0.56401

➔ RandomForestClassifier

The Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then It collects the votes from different decision trees to decide the final prediction.

Training accuracy Score : 0.99951

Validation accuracy Score : 0.57883

Multiclass Classification

➔ XGBoost classifier

XGBoost, which stands for Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library. It provides parallel tree boosting and is the leading machine learning library for regression, classification, and ranking problems.

Training accuracy Score : 0.49460

Validation accuracy Score : 0.47351

➔ Support vector machine classifier

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model sets of labeled training data for each category, they're able to categorize new text

Training accuracy Score : 0.89697

Validation accuracy Score : 0.60762

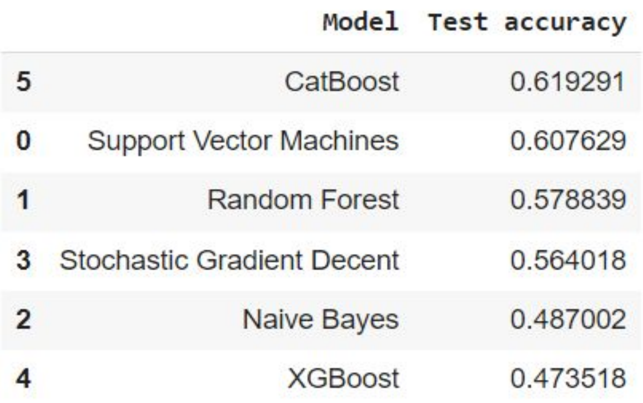
➔ CatBoost classifier

CatBoost is based on gradient boosted decision trees. During training, a set of decision trees is built consecutively. Each successive tree is built with reduced loss compared to the previous trees. The number of trees is controlled by the starting parameters.

Training accuracy Score : 0.66894

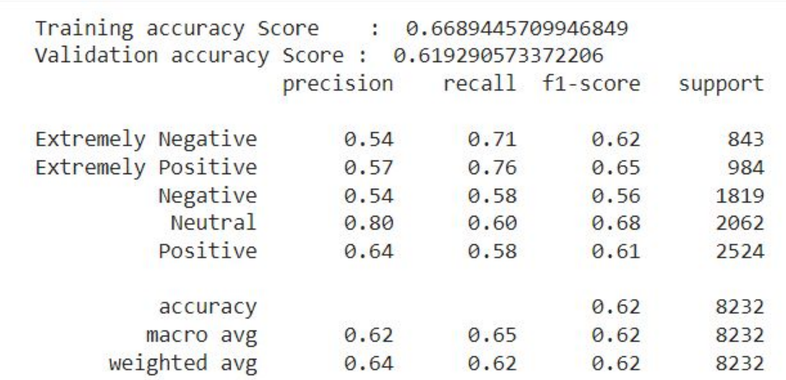
Validation accuracy Score : 0.61929

Multiclass Classification Results

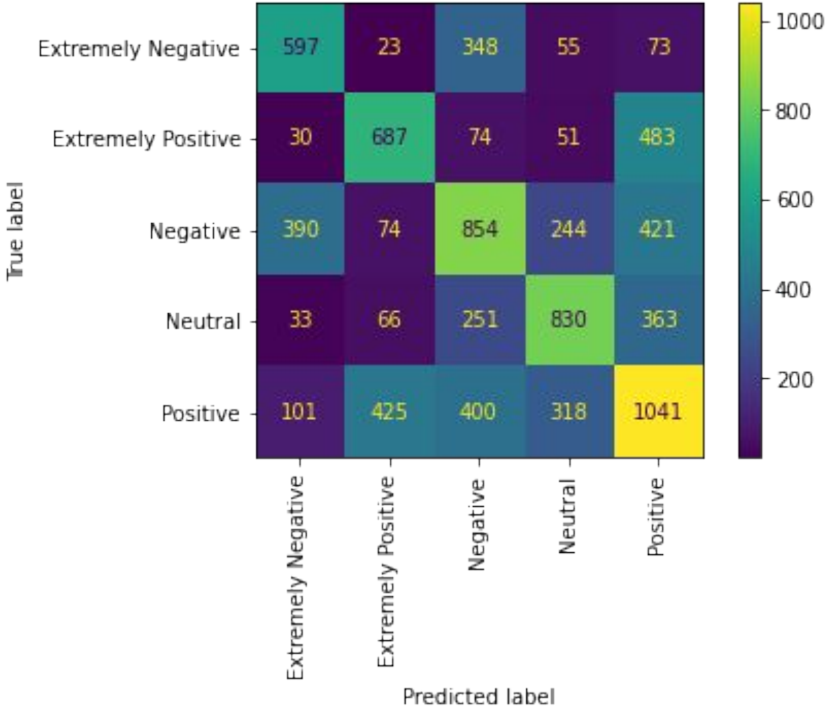


	Model	Test accuracy
5	CatBoost	0.619291
0	Support Vector Machines	0.607629
1	Random Forest	0.578839
3	Stochastic Gradient Decent	0.564018
2	Naive Bayes	0.487002
4	XGBoost	0.473518

CatBoost Classifier



```
Training accuracy Score    :  0.6689445709946849
Validation accuracy Score :  0.619290573372206
                    precision    recall  f1-score   support

Extremely Negative       0.54      0.71      0.62       843
Extremely Positive       0.57      0.76      0.65       984
          Negative       0.54      0.58      0.56      1819
           Neutral       0.80      0.60      0.68      2062
          Positive       0.64      0.58      0.61      2524

          accuracy                           0.62      8232
         macro avg       0.62      0.65      0.62      8232
      weighted avg       0.64      0.62      0.62      8232
```



Binary Classification Models

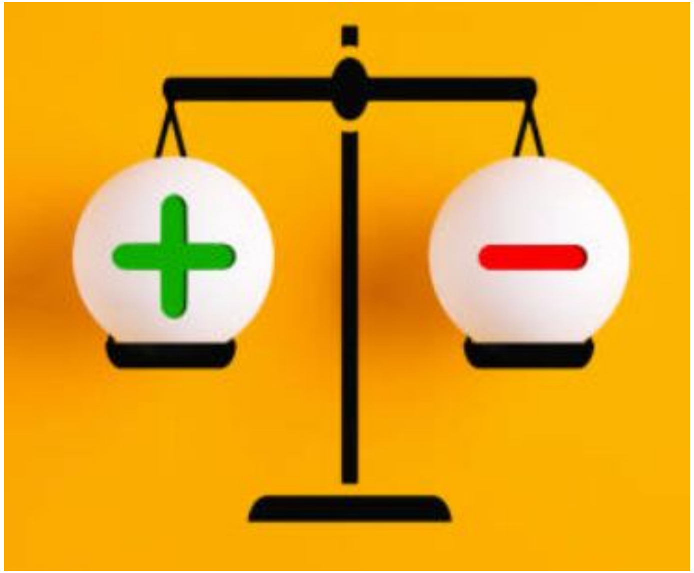
Target Classes

1. Extremely Positive
2. Positive
3. Neutral

Positive

1. Extremely Negative
2. Negative

Negative



Binary Classification Models

➔ Voting Classifier

A voting classifier is a machine learning estimator that trains various base models or estimators and predicts on the basis of aggregating the findings of each base estimator. The aggregating criteria can be combined with a decision of voting for each estimator output.

Training accuracy Score : 0.93542
Validation accuracy Score : 0.86382

➔ Stacking Classifier Algorithms

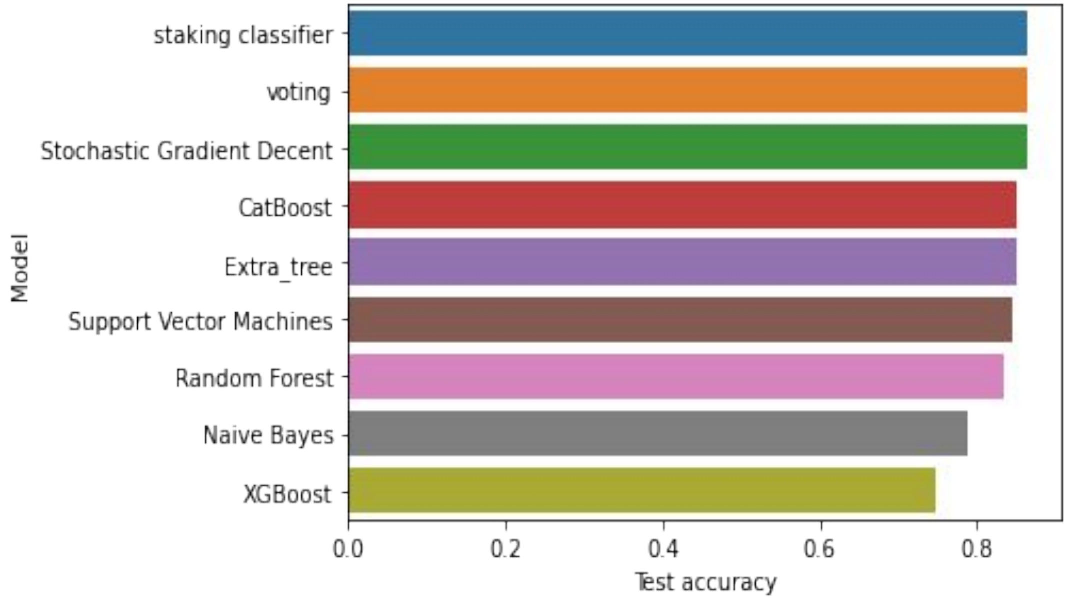
Stacking is a way of ensembling classification or regression models; it consists of two-layer estimators. The first layer consists of all the baseline models that are used to predict the outputs on the test datasets

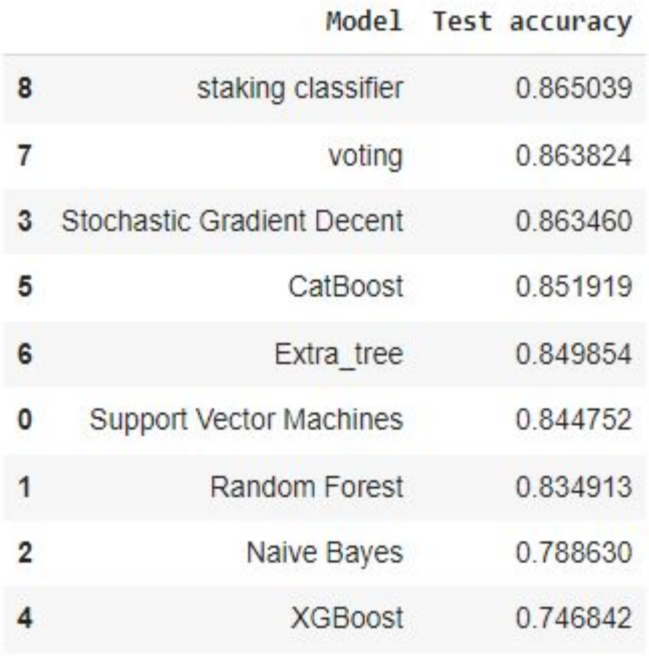
Training accuracy Score : 0.99429
Validation accuracy Score : 0.86503

Hyper Parameter Tuning

```
 # vectorization
 vectorizer = CountVectorizer(decode_error = 'replace',stop_words =
 stop,max_features=8000)


 # sgd classifier
 SGDClassifier(loss = 'hinge', penalty = 'l2', random_state=0

 # stacking
estimators=[('sgd', sgd_clf), ('catboost', clf2), ('extra_tree', extra_tree)]

# voting classifier
voting = VotingClassifier(estimators=[('sgd', sgd_clf), ('catboost', clf2), ('extra_tree',
extra_tree)],voting='hard')
```

Binary Classification Models Result





	Model	Test accuracy
8	staking classifier	0.865039
7	voting	0.863824
3	Stochastic Gradient Decent	0.863460
5	CatBoost	0.851919
6	Extra_tree	0.849854
0	Support Vector Machines	0.844752
1	Random Forest	0.834913
2	Naive Bayes	0.788630
4	XGBoost	0.746842

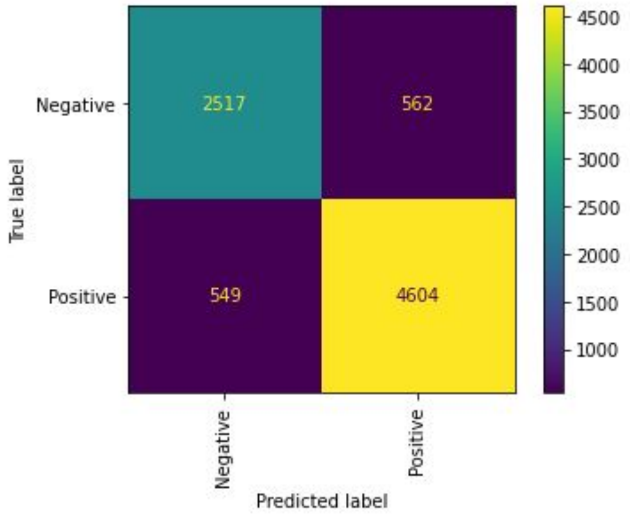
Best Classification Model

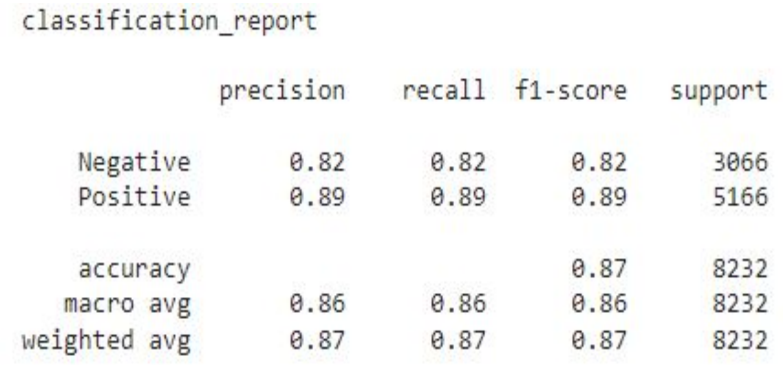
➔ Stacking Classifier Algorithms

Stacking is a way of ensembling classification or regression models; it consists of two-layer estimators. The first layer consists of all the baseline models that are used to predict the outputs on the test datasets

Training accuracy Score : 0.99429
Validation accuracy Score : 0.86503

confusion_matrix





```
classification_report

              precision    recall  f1-score   support

    Negative       0.82      0.82      0.82      3066
    Positive       0.89      0.89      0.89      5166

    accuracy                           0.87      8232
   macro avg       0.86      0.86      0.86      8232
weighted avg       0.87      0.87      0.87      8232
```

Conclusion

- ➢ We have tried Stacking classifier, Stochastic Gradient Descent Classifier, voting, CatBoost, Extra Tree, Support Vector Classifier, Random Forest Classifier, Multinomial Naive Bayes and XGboost classifier
- ➢ We have Extracted new Features as number of words, characters and sentences to check sentiment of the Tweet.We can see that if the number of words, characters and sentences are more than sentiment of the tweet is more positive. Neutral Sentiment Tweets consist of fewer words, characters and sentences. After evaluating models with these features we have found that these features are not contributing much in the model performance. Hence we have decided to drop them in Binary classification models
- ➢ Firstly we evaluate the multi class models as categories -Positive, Extremely Positive, Neutral, Negative, Extremely Negative. Most of the people (28%) were having Positive sentiment about covid followed by Negative (24%), Neutral(19%), Extremely Positive (16%) and Extremely Negative(13%). Our Target variable is not Unbalanced as all Categories are not having much differences between them.
- ➢ The performance of the Multiclass Models is not satisfactory then we convert the problem into binary class. We have kept only two Sentiments as Positive and Negative.

Conclusion

- ➢ The Binary Stacking Classifier has the best performance of accuray 0.8650 followed by CatBoost and Extra Tree classifier.
- ➢ We have checked for Voting classifiers and Stacking Classifiers with hyperparameter tuning.
- ➢ For vectorizing we have tried all kinds of vectorizing methods i.e. tfidfvectorizer ,countvectorizer, ngrams, after doing hyper parameter tuning, we have decided to use countvectorizer for vectorization.
- ➢ Sentiment Analysis is done based on Positive and Negative Sentiment.
- ➢ Feature Importance is calculated based on Most Frequent words in each class.
- ➢ We can see London, United States, New York, Washington DC, United Kingdom ,India, Australia, USA are the top locations as far as count of tweets are concerned.
- ➢ We can see that maximum tweets were done on 20 march 2020, when the first lockdown was declared. People were more active on Twitter in the month of March because it was the early stage of Corona virus Pandemics and people wanted to know more about this disease.

Challenges Faced

1. As Data is related to natural language processing sentiment analysis, Time required for data pre processing is high.
2. Feature Extraction and Computation Cost is high.
3. There is a wide scope for Vectorization techniques.
4. Algorithms like SVC, XGBoost, CatBoost have high computation cost.
5. Time required to run Stacking and Voting Algorithm more than other algorithms

Scope

As we are dealing with sentiment analysis of coronavirus tweets, It's very important to classify sentiment as either positive or negative to use it as reference for different stakeholders.

Governments can make use of this information in policymaking as they are able to know how people are reacting to this new strain, what all challenges they are facing such as food scarcity, panic attacks, etc. Various profit organizations can make a profit by analyzing various sentiments as one of the tweets tells us about the scarcity of masks and toilet papers.

These organizations are able to start the production of essential items thereby making profits. Various NGOs can decide their strategy of how to rehabilitate people by using pertinent facts and information.We could do the analysis with three classes as positive, negative and neutral.

Thank You