Begüm Arslan

FINAL PROJECT REPORT

SENTIMENT ANALYSIS of COVID19 TWEETS

PART1: Scraping Tweets

With the help of the snscrape module, COVID19 tweets from '2021-01-01' to '2022-01-01' were extracted with the content of date and content. The result was transformed into a data frame and the resulting data frame was transformed into a csv file named 'covid_tweets_2021.csv'.

PART2: Preprocessing

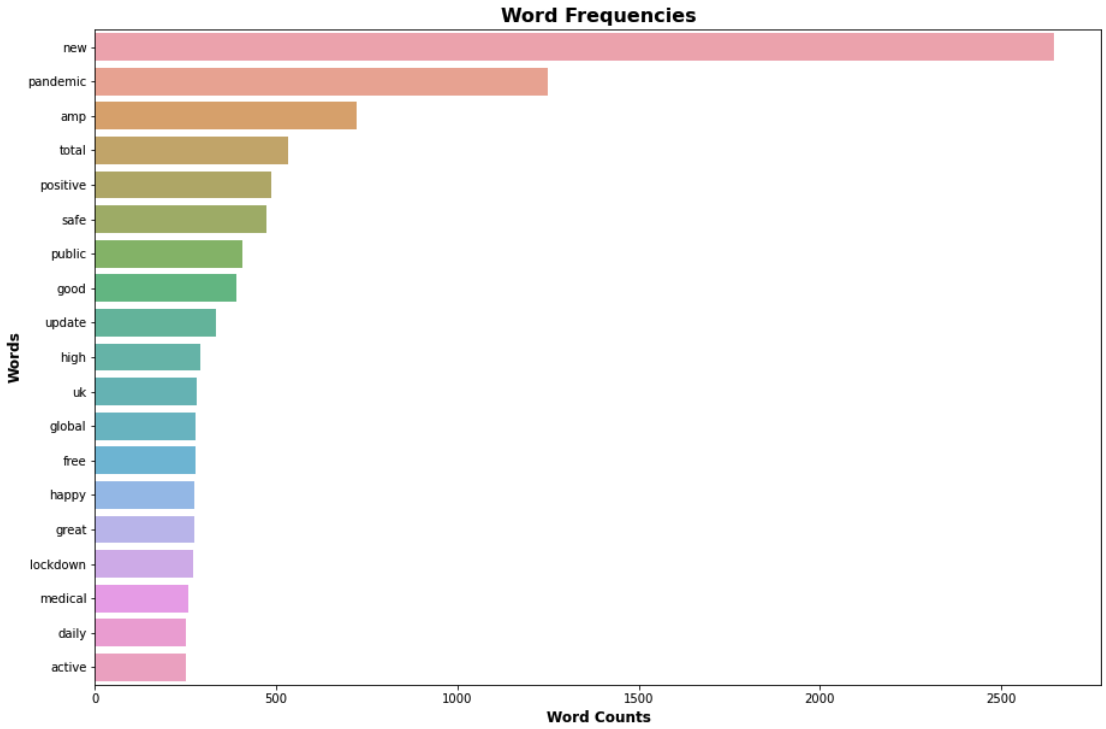
Cleaning data is an important task of NLP projects in order to models perform accurately. For texts, it is important to leave the text with its pattern in order to model perform well. First, the extra spaces, punctuations, low frequency words were removed from the tweet column. All the letters were also lower-cased. In order to remove English stop words, two method was applied. The first one is nltk corpus' stop words method and the other method is the spacy library. After removing the stop words, tokenization applied which big texts divided into its smaller parts called tokens. These tokens are what created the pattern inside the tweets.

PART3: Language Processing with BERT

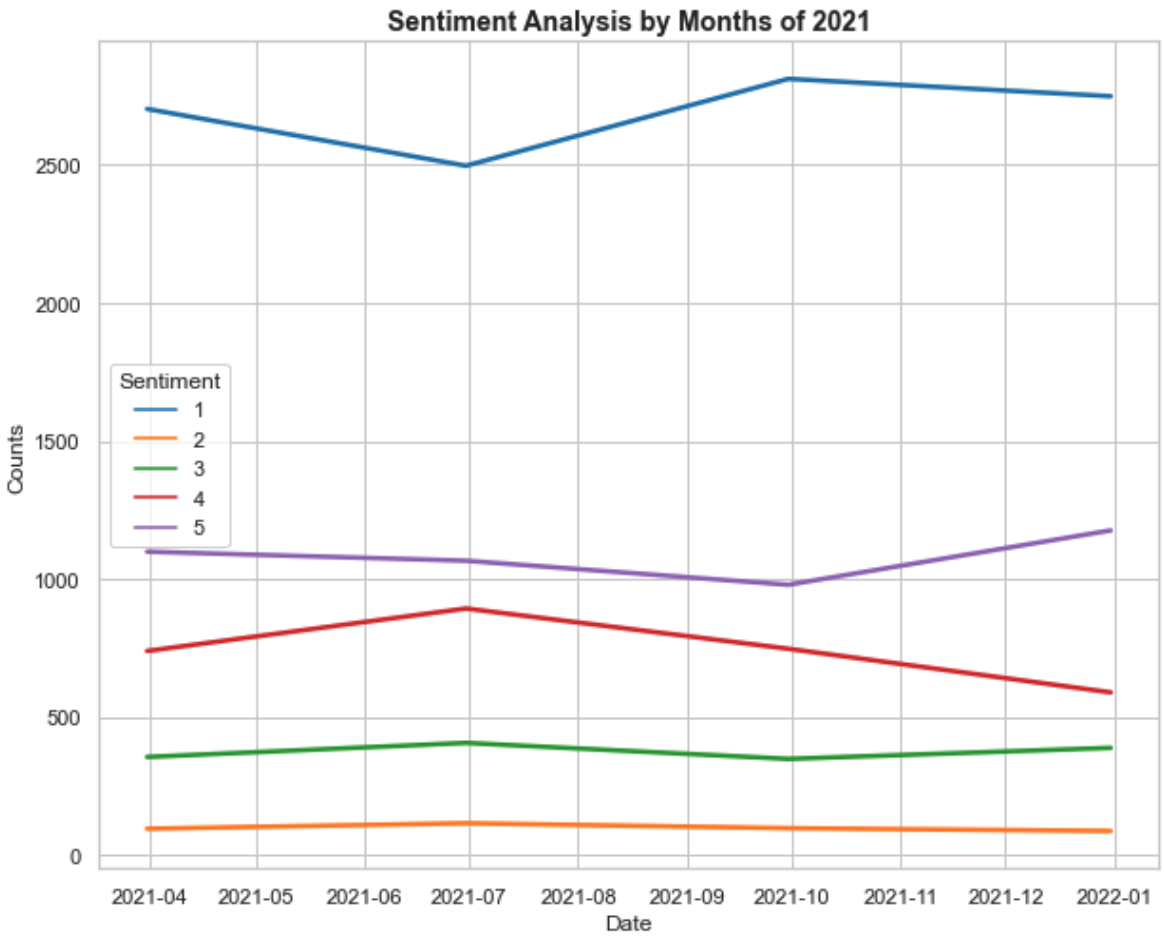
BERT is a transformer type which outperforms any other tool in nlp techniques. It is also a masked language model which allows contextual word embeddings. Since it is a pretrained model, it would be beneficial to use it as original sentiment of the words. Thus, at Part 3, BERT transformer was used in order to label the sentiments of the text. The labels of the BERT ranges from 1 to 5, which 1 represents the most negative sentiment and 5 represents the most positive sentiment.

PART4: Data Analysis

In order to understand what the contents of the tweets are, data analysis was performed. The most frequent occurred words analyzed throughout the whole dataset and the first 20 frequent words were outputted as follows:



Then, time series analysis was performed and showed that negativity of tweets decreased as mid of 2021, however it started to increase again. Nowadays, it is decreasing again, but not in a fast manner. Also, it can be seen that there is an imbalanced dataset which negative sentiments are the dominant ones.



Finally, what words dominated the positive sentiments and negative sentiments were analyzed and the outputs are as below:

Frequently Used Positive Words

Frequently Used Negative Words

As it can be seen, among positive words the most frequently used ones are: Safe, Great, Happy, Free, Good, Important, Live…. Among negative words the most frequently used ones are: Total, positive, public, lockdown and uk.

PART5: MODELLING

3 Methods were used in order to compare the model performances. The data with labels which output from BERT is loaded. The ‘text’ column of tweets were first transformed into token

counts and then, TF-IDF representations. Then, 2 models were applied: 1) Random Forest Classifier, 2) SVM

Accuracy of RFC: 0.668

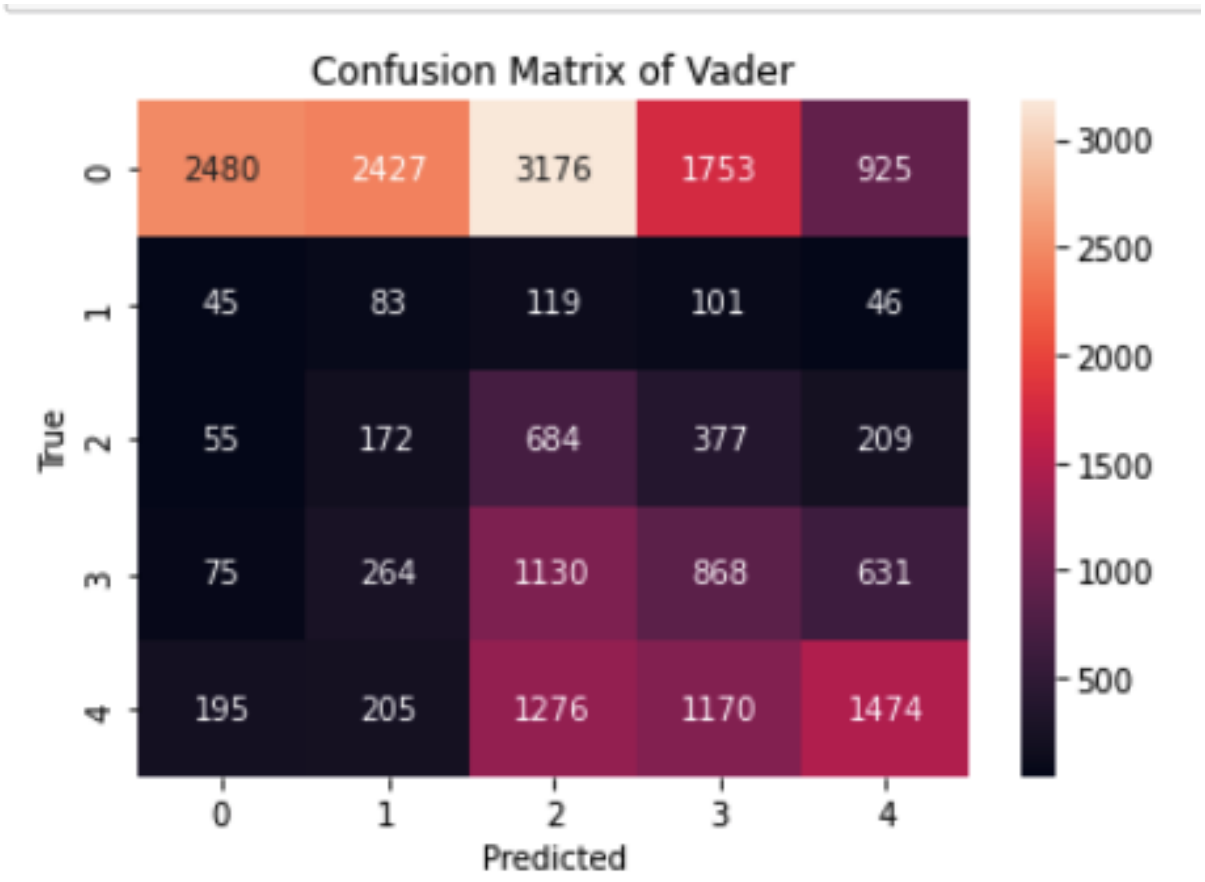
F1 score of RFC: 0.629

Accuracy of SVM: 0.711

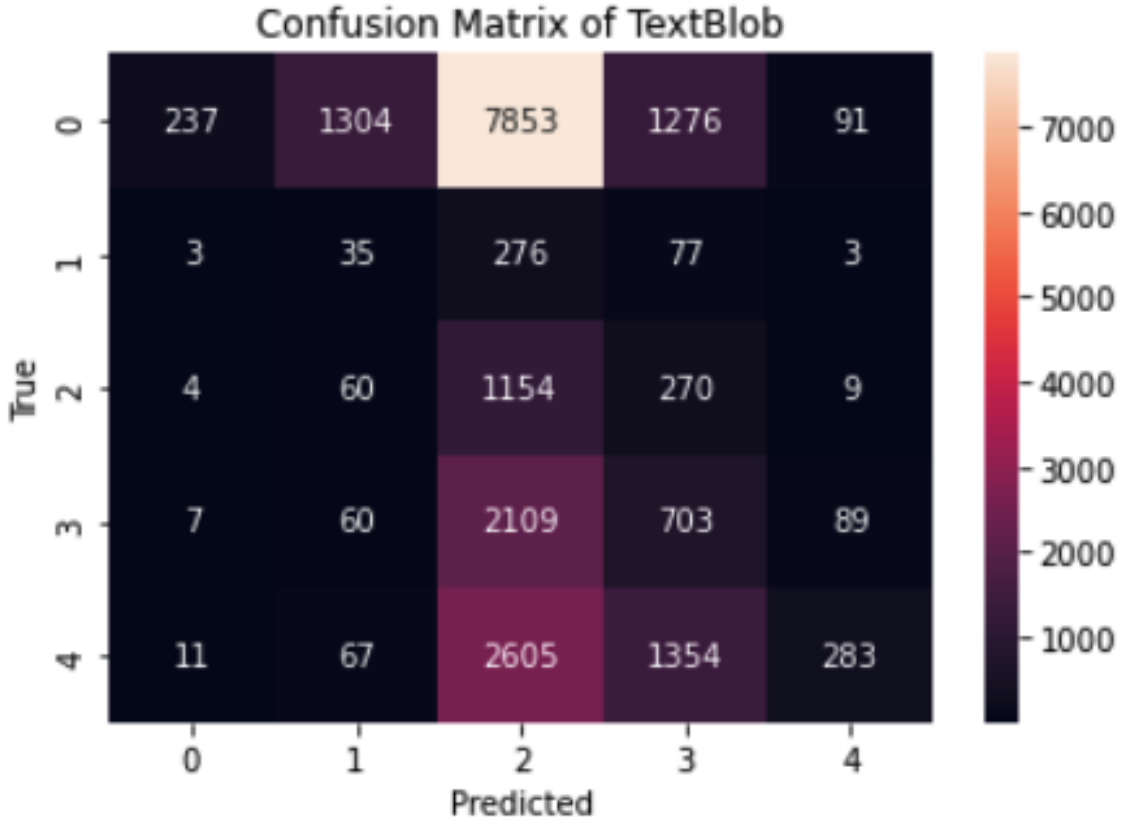
F1 score of SVM: 0.692

SVM performed better than Random Forest Classifier. Since the dataset is imbalanced, F1 score is also an important factor. F1 score of SVM also better than Random Forest Classifier. When looked at confusion matrix of SVM, it can be seen that SVM performs well in classifying the edge scores, however it cannot perform well at classifying points 4.

Another method was the VADER (Valance Aware Dictionary Sentiment Reasoner) method. This method is focused only to sentiment analysis. According to its polarization score outputs, since BERT sentiments contained 5 classes, the output was divided into 5 sets. However, VADER method did not perform well as well as SVM which it output accuracy score of 0.28 and F1 score of 0.330.



The last method used was TextBlob method. This method is built on top of NLTK library. It scores for polarity, subjectivity and intensity. However, it did not perform as well as SVM and VADER which it gave accuracy of 0.121 and F1 score of 0.092.



When VADER compared with TextBlob method, it gave better predictions, however it can also be seen that VADER gave inconsistent inaccurate results at more different categories. As a result, it can be stated that the best method is modelling TFIF vectorized tweets with SVM, which is a feature-based method. It can be seen from the analyses that feature-based methods such as RFC, SVM outperformed rule-based methods such as TextBlob and VADER.