Capstone Project Submission

Instructions:

Name, Email and Contribution:
Name - Potdar Vinayak **Email - vinayak.potdar.vp@gmail.com** **Contributions:** **Data Wrangling** **Null Value Analysis and Treatment** **Exploratory Data Analysis** **Data Preprocessing** **Stemming** **Vectorization** **Model fittings and Performance Analysis** **Model Selection.**
Please paste the GitHub Repo link.
Github Link:- https://github.com/vinayakpotdar2114/Corona-Virus-Tweet-Sentiment-Analysis
Please write a short summary of your Capstone project and its components. Describe the problem statement, your approaches and your conclusions. (200-400 words)
Problem Statement This challenge asks you to build a classification model to predict the sentiment of COVID-19 tweets. These tweets have been pulled from Twitter and manual tagging has been done then. **Data Description** The dataset consists of web scrapped data, from Twitter. The dataset consists of tweets made by users regarding Covid-19. We can see records from year 2020. The names and usernames have been given codes to avoid any privacy concerns. The provided data does not consist of many features. The features important for sentiment analysis are just two. Although we will be needing the rest of the features for the Exploratory Data Analysis. **Approach** We wanted to predict the sentiments based on the tweets that were done. In order to do so, we need to work with the keywords that describe the sentiment. First we did some exploratory data analysis to understand if there is any relation of features with the sentiment other than the tweets themselves. After getting all the insights from the dataset through EDA, we proceeded with 'Data Preprocessing', where we model the dataset so that it can be fed to the models. The tweets are the input for this process. Here we started to remove irrelevant data from the tweets such as urls, tagged usernames, punctuations, special characters and stop-words.

After removing the irrelevant data from the tweets, we needed to remove multiple forms of same words, so as to reduce redundance. Vectorization is a process which will convert each word from the sentence into a feature. Here we don't need multiple forms of the same words e.g.- family and families. In order to achieve this we need to do either stemming or lemmatization. We went ahead with stemming.

Now that we had a desired workable dataset, we tried to fit four different learning models along with hyper-parameter tuning. We did the same process again after combining two classes and reducing them from 5 to 3.

Conclusion

Here, we conducted two experiments with 5 classes and 3 classes. The first was conducted with hyperparameter tuning and Catboost turned out to be the best performing model with 0.8069 testing accuracy. In the second scenario we got Catboost as the best perfroming model again. Here SVM improved its performance drastically as compared to the first scenario.