**Corona Virus Tweets Sentiment Analysis using ML Approach**

|  |
| --- |
| **Team Member’s Name, Email and Contribution:** |
| Name: Debanjan Ganguly  Email: [officialdeba10@gmail.com](mailto:officialdeba10@gmail.com)  Project Contribution:   * Data Exploration * Basic Text Preprocessing * Exploratory Data Analysis * Text Normalization * Model Preparation – Defining Pipeline, Hyperparameter Tuning, Evaluation of Models using Metrics used for Classification * Model Selection and Predict Sentiment |
| **GitHub Repo & Drive Link:** |
| GitHub Link: - <https://github.com/awesomedeba10/Coronavirus-Tweets-Sentiment-Analysis---AlmaBetter-Capstone-Project>  Drive Link: - <https://colab.research.google.com/drive/19JBAqC_ZVS8nnbkRemPwXzK3_TNzguKM?usp=sharing> |
| **Project Summary:** |
| Natural Language Processing (NLP) allows machines to break down and interpret human language. NLP combines the power of linguistics and computer science to study the rules and structure of language, and create intelligent systems (run on machine learning and NLP algorithms) capable of understanding, analyzing, and extracting meaning from text and speech.  Sentiment analysis (or opinion mining) is a natural language processing (NLP) technique used to determine whether data is positive, negative or neutral. Sentiment analysis focuses on the polarity of a text (positive, negative, neutral) but it also goes beyond polarity to detect specific feelings and emotions, depending on how one wants to interpret user feedback and queries.  In our project, we had performed Aspect-based Sentiment Analysis where our aspect-based classifier would be able to determine that whether a comment or tweet expresses a negative or positive opinion, or it is just a neutral statement. Our classification will be a multiclass one, where we have over 30 thousand corona virus related tweets of 5 classes distributed within Extreme Positive to Extreme Negative.  We’ve started by some rule-based approaches of traditional NLP, where we will lower case all alphabets, replace twitter mentioned user, any links, any non-compliance characters of utf-8 charset, remove stop words, short irrelevant words and perform lemmatization or stemming. Then we’ve plotted some graph to show sentiments over time, most trending hashtags, proportions of sentiments, most common words for each sentiment class. Next to it, text vectorization was done using bag-of-words approach and data is split into training set and testing set and training was done after.  We’ve worked with [scikit-learn](https://scikit-learn.org/) library and used only statistical models like Naïve Bayes, Support Vector Machines, Decision Tree available in scikit learn library. Hyper parameter tuning has also been performed along with a approaches to reduce classes for better accuracy and random sampling has also been done using [imblearn](https://imbalanced-learn.org/stable/) library.  We’ve achieved a maximum of **avg. weighted f1 score** of **0.7868** and **balanced accuracy score** of **0.7775** using balanced weighted linear support vector classifier. Various metrics like balanced accuracy, roc-auc graphs, most influential words have also been plotted. In the end, a prediction function is created using the best performed model which will essentially return the sentiment and confidence score (using decision function).  The results we obtained are marginal. We suspect if we get more data and then try to balance the classes, that will reduce the bias attached. Also, we haven’t used any ensemble techniques, normally ensemble techniques tend to provide better performance with exchange of high computation cost. Also, there is future scope to use either any ensemble techniques or deep learning techniques like performing word embedding and then training with Keras Api, the we can expect a better performance. |
| **Reference:**  *Declaration: The references mentioned below are not the direct copy of code but used for learning purpose of various techniques or metrics or ideas that have been adapted in this project after learning and understanding the concepts thoroughly from referenced articles.*  <https://towardsdatascience.com/comprehensive-guide-to-multiclass-classification-with-sklearn-127cc500f362>  <https://www.techscience.com/cmc/v69n1/42767/html>  <https://www.kaggle.com/code/venkatkrishnan/nlp-multiclass-onevsrestclassifier/notebook>  <https://towardsdatascience.com/having-an-imbalanced-dataset-here-is-how-you-can-solve-it-1640568947eb>  <https://medium.com/@aneesha/visualising-top-features-in-linear-svm-with-scikit-learn-and-matplotlib-3454ab18a14d>  <http://michael-harmon.com/blog/NLP1.html> |