**ASSIGNMENT REPORT**

Text Classification is an automated process of classification of text into predefined categories. We can classify Emails into spam or non-spam, news articles into different categories like Politics, Stock Market, Sports, etc.This can be done with the help of Natural Language Processing and different Classification Algorithms like Naive Bayes, SVM and even Neural Networks in Python.

**STEP -1** :**Add the Required Libraries**

The following libraries will be used ahead in the article.

Pandas, numpy, sklearn, collections

**STEP -2: Load the datasets Data pre-processing**

Datasets: train.csv and valid.csv

Drop any null values

**STEP -3: Word Vectorization**

It is a general process of turning a collection of text documents into numerical feature vectors.Their are many methods to convert text data to vectors which the model can understand but by far the most popular method is called TF-IDF. This is an acronym than stands for “*Term Frequency — Inverse Document*” Frequency which are the components of the resulting scores assigned to each word.

* **Term Frequency**: This summarizes how often a given word appears within a document.
* **Inverse Document Frequency**: This down scales words that appear a lot across documents.

Finally we will transform*train\_x*and *valid\_x*to vectorized*Train\_X\_Tfidf* and *Test\_X\_Tfidf*. These will now contain for each row a list of unique integer number and its associated importance as calculated by TF-IDF.

**STEP -4: Use the ML Algorithms to Predict the outcome**

1. Naive Bayes Classifier Algorithm
2. Support Vector Machines Algorithm

**STEP -5: Find Accuracy, F1-Score, Precision Score and Recall Score**

Naive Bayes Accuracy Score -> 74.98400511836213

F1 Score -> 0.5144887221693434

Precision Score -> 0.5035984260738043

Recall Score -> 0.7173757203321437

SVM Accuracy Score -> 94.72168905950096

F1 Score -> 0.8545557994526632

Precision Score -> 0.8505615195096956

Recall Score -> 0.8612970344375406