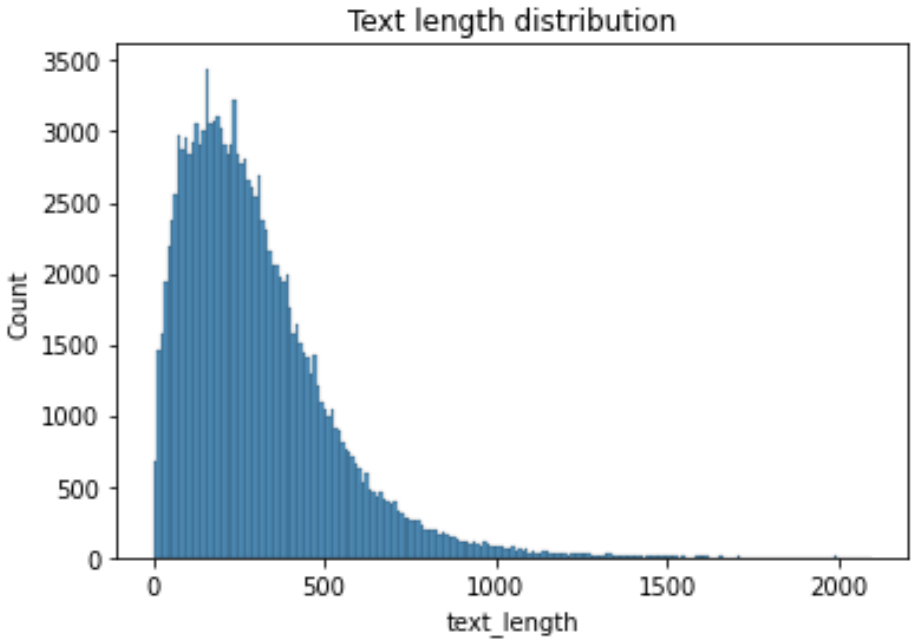
COL772 Assignment 1

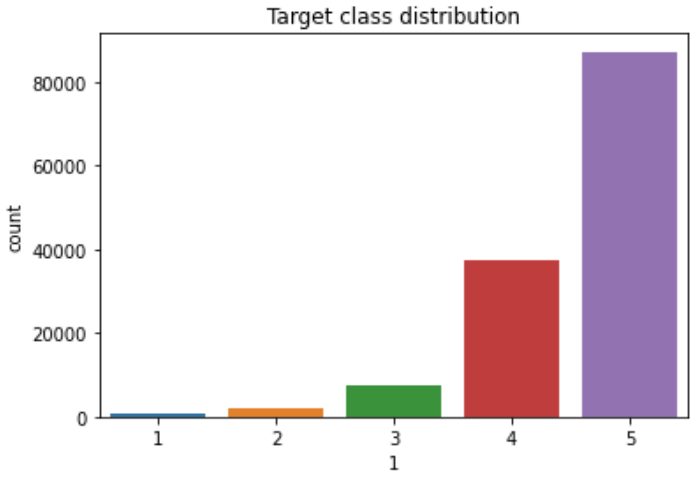
Rishav Kumar Rajak (2018CH70302)

First of all I have done some exploratory data analysis of the data. There are 44 null value entries, since it is pretty small value in comparison to the total dataset. I removed these entries, so total number of data we have left with is 134660. Below are some of the plots that I have used to visualize the data.



Above histogram shows the reviews length distribution. We can see that most of the text size lies between 0-500 length. The below image shows the most common word in the dataset.

I have also plotted the class distribution of the data which is depicted by below image.



The above graph is skewed, which shows pure case of class imbalance. This need to be taken care while training to get good accuracy. Various techniques can be implemented like resampling, class weights, SMOTE. In this assignment I have used class weights to give more emphasis to least dominant classes.

I have included the code for the above analysis except for the most common words. Because it uses wordcloud through pip install.

Model Specifications

I have selected SVM Classification model for this assignment. I have also tried Naïve Bayes Algorithm, but the SVM was performing quite better in my case. I have used the weighted F1 score as stated in the problem statement. For the initial runs of the model, I have used stratified train, test split with a ratio of 0.2. But in the final updated model, I have included K-fold cross validation. I have performed smart grid search(Randomized Grid Search), in order to find the best parameters of the model. I have used **RandomizedSearchCV** class from scikit-learn to perform the smart grid search. The **param_dist** dictionary defines the parameter distributions for the search.

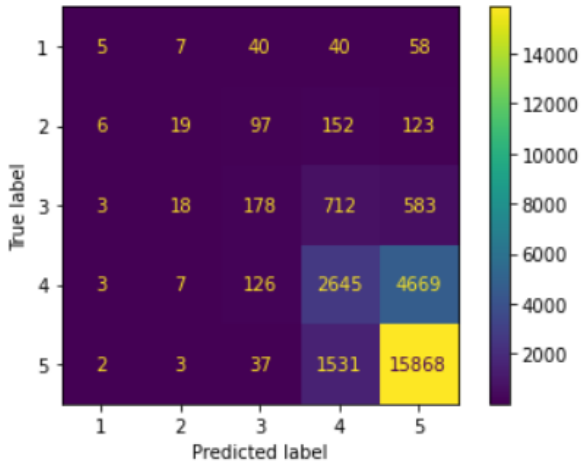
```
param_dist = {'C': np.logspace(-3, 3, 7),
              'kernel': ['linear', 'rbf'],
              'dual: [True, False],
 'tol': [1e-3, 1e-4, 1e-5]}
```

The best parameters were found out be C: 0.1, kernel='linear', dual='False' and tolerance='1e-4'.

Below are different model results (including train, test F1 score and confusion matrix)

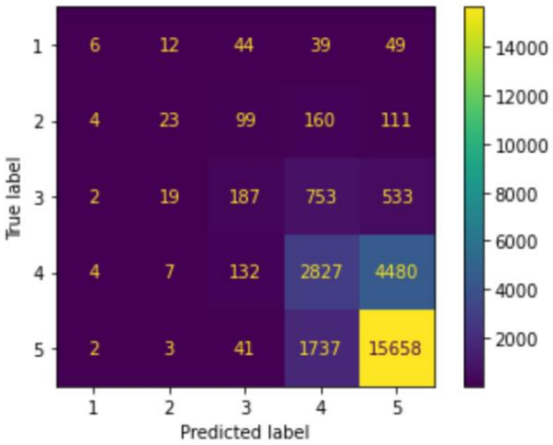
1. Baseline Model: (Only Vectorizer is used)

Training SVM F1_Score: 0.738961568527477
Testing SVM F1_Score: 0.6127754493991758



2. Class Weight is used for dealing with class imbalance. In SVC Model, class_weight was set to balanced

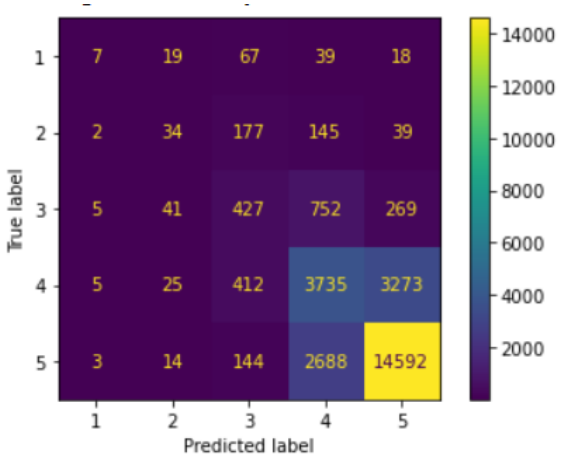
Training SVM F1_Score: 0.7918609597307505
Testing SVM F1_Score: 0.6595740072650077



Test set accuracy is increased by introducing class weights as we have skewed class distribution.

3. After Lemmatization, Stop words removal, Tf-ID Weighting:

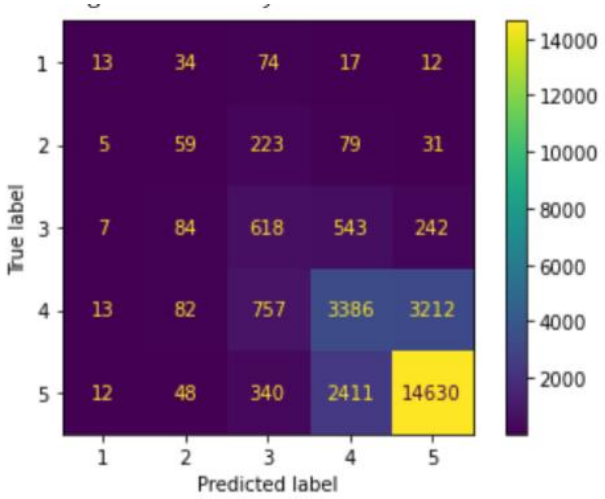
Training SVM F1_Score: 0.9918422411105323
Testing SVM F1_Score: 0.689694653821937



The accuracy has increased upon Lemmatization, Stop words removal, Tf-ID Weighting.
Also the time taken to train the model has significantly increased due to these preprocessing.

4. After Part of Speech Tagging, bigrams, trigrams, and capitalization information and including all the ab ove features.

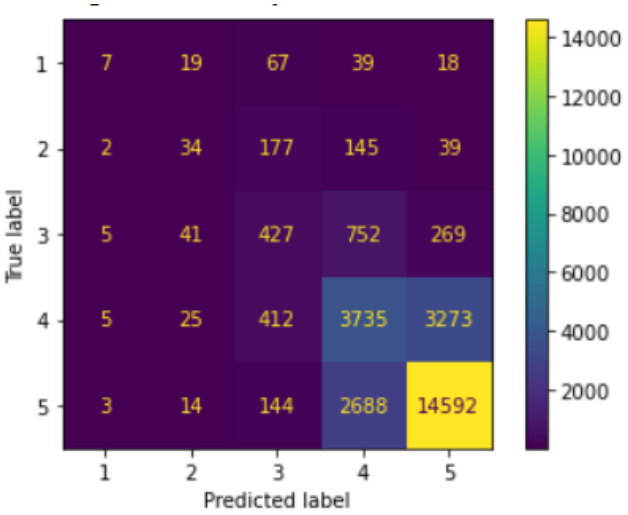
Training SVM F1_score: 0.8348965891094389
Testing SVM F1_score: 0.6903988881133809



The accuracy has slightly increased upon POS Tagging, using Unigrams, Bigrams and trigrams.

5. After selecting the best parameters from Randomized Grid Search, apply 5-fold cross validation. I got t he below results on the validation set.

Average accuracy: 0.6867515456972496
Best accuracy: 0.69918440302383738



I have saved the tfIDVectorizer and best model file as a pickle file in trained_model folder.