Name: Rahul Vijayvargiya
Student: 245784

Task:
Sentiment Labelled Sentences
Link for dataset:

[Kaggle Link to Dataset]

The collection contains sentences derived from the opinions of website users.
The dataset is used to learn models that recognise sentence sentiment between two classes – positive and negative. The dataset is designed for the classification task.

Sol.

Hello, let me tell you about my project, it's based on sentiments, we are using supervised learning technique called classification here,
with the help of sklearn Naive Bayes and Random Forest We gonna Classify which sentiment is good or which one is negative, we are using matplotlib, sklearn and pandas for our small project

Naive Bayes:

The Naive Bayes classification algorithm is a probabilistic classifier. It is based on probability models that incorporate strong independence assumptions.
The independence assumptions often do not have an impact on reality. Therefore they are considered naive.

Random Forest Classifier:

The random forest is a classification algorithm consisting of many decision trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree

We have a Dataset for reviews from yelp, amazon, imdb, we have to train our machine learning model to predict the sentiments of the sentences whether it is bad or good

Since it’s a supervised learning,

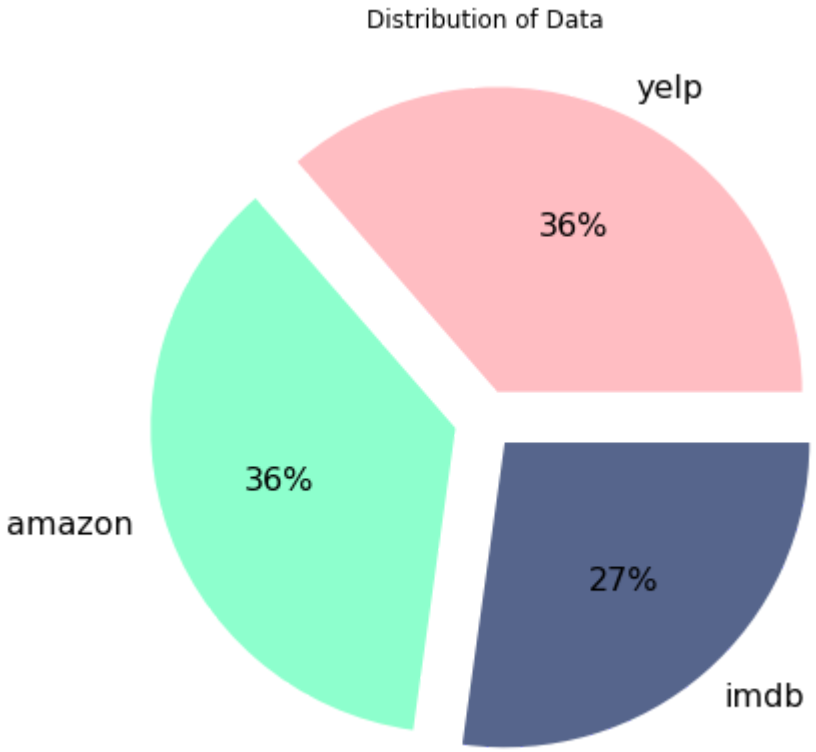
1. The dataframe of our data set, it has label, source, sentence,
   Where label tells us about 1 is positive and 0 stands for a bad and negative review

2. we are converting our dataframe column review into lower case, we have to feed clean data into our model
   For e.g. Hate, hate and HATE, haTE are not the same

3. Furthermore we are removing all the punctuation marks from the columns

4. Removing stop words doesn't add any value all to the sentence
   e.g. I, you, is are the stop words and doesn't add any value
   From sklearn we are using count vectorizer to remove stop words such as top words in English are **"a", "the", "is", "are"** and etc

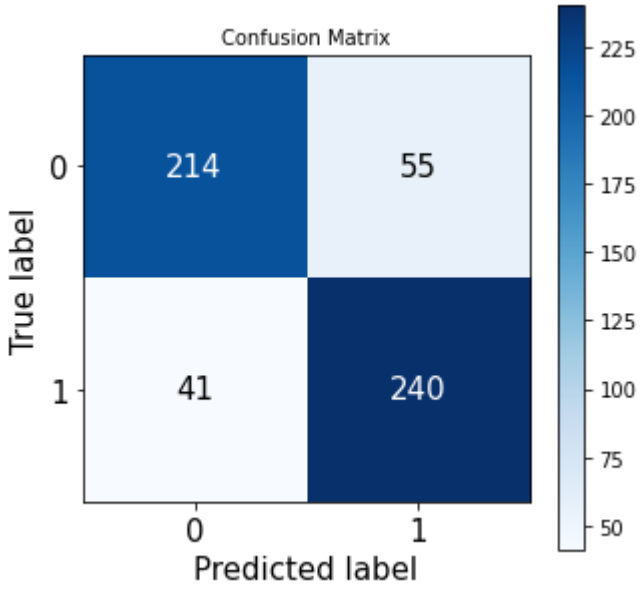
5. Pie plot to show distribution of dataset



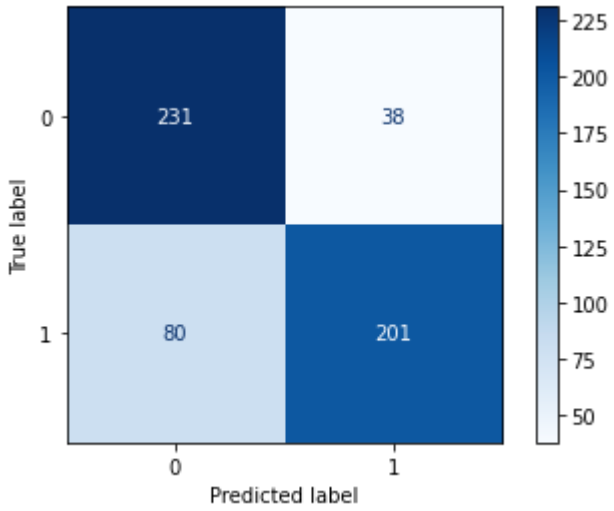
   As you can see the fair share of data imdb shares the lowest percentage

6. After removing the stop words and removing the punctuation, lowercase the dataframe,
   We gonna convert our data into a word vector using countvectorizer from sklearn

Later, We are splitting data into training and test, and importing multinomial naive bayes model from sklearn
And fitting our training data into model
With the help of naive bayes classifier we predict, where alpha = 2
454 sentences classified correctly, 96 sentences classified incorrectly



With Random Forest:



7. Test data accuracy of the Naive Bayes Multinomial model is 82.55%

We also used random forest classifier, with the help of random forest classifier we predicted 436 correct and 114 incorrect sentences and
Test data accuracy of model is 79.27%

8. Test case,

```
example = ['i hate you',
           'i love you',
          'i hate america',
          'i hate everyone',
          'product is bad',
          'not all bad',
          'boom boom hate',
          'this new movie ']
```

```
Predicted output = array([0, 1, 0, 0, 0, 0, 0, 1]
Classified all the test correctly on both Naive Bayes and Random
Forest Classifier
```

9. Cross Validation using K-FOLD and GRIDSEARCHCV

   KFOLD:

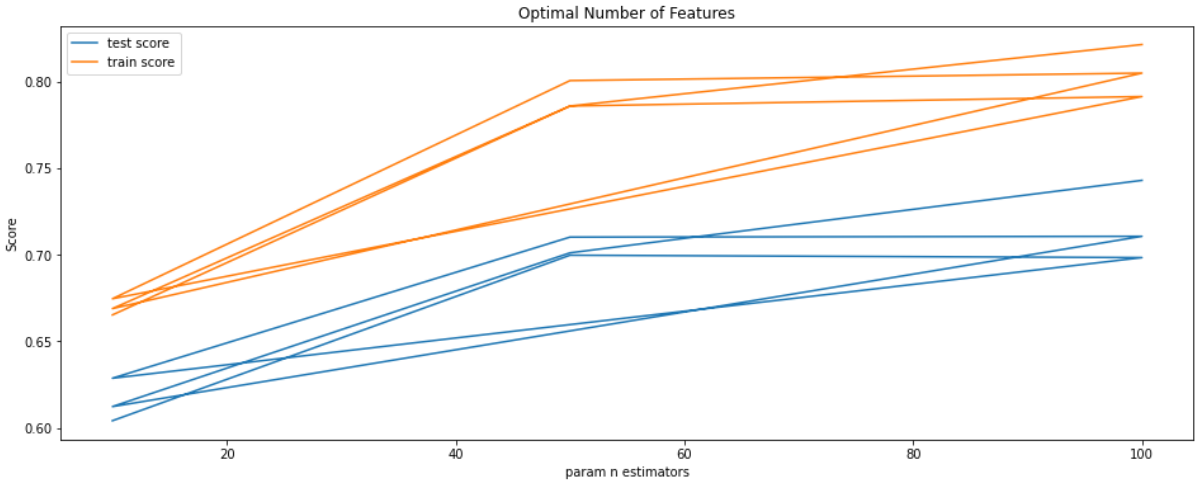
   Using 5 folds on Naive Bayes gave us average accuracy of 78.16%
   Using 10 folds on Naive Bayes gave us average accuracy of 78.75%

   GridSearchCV:

   I perform GridSearchCV method on Random Forest Classifier turned out the best parameter for our model is:

```
{'criterion': 'gini',
 'max_depth': 9,
 'max_features': 'auto',
 'n_estimators': 100}
```

   With this parameter i achieve mean accuracy of 72%



Experiments:

Experiments using both Naive Bayes and Random Forest Classifier, Tweaking Hyper Parameter such as laplace smoothing in naive bayes and number of tree and depth in random forest and gini and entropy as criterion, using TF-IDF Vectorizer we i saw change in model accuracy, decreasing both random forest and naive bayes, changes in accuracy in was not much:

The Accuracy Score on test data using random forest classifier 78.55%

The (testing) accuracy of the Naive Bayes model is 81.64%

5 Fold Mean Accuracy on Naive Bayes with Alpha=3 is = 78%
10 Fold Mean Accuracy with Alpha = 2 on Naive Bayes is = 78.75&

Best parameter in random forest so far is the same in GRIDSEARCHCV:
Provide the better accuracy on TF-IDF Vectorizer is = 82.1314

```
{'criterion': 'gini',
 'max_depth': 9,
 'max_features': 'auto',
 'n_estimators': 100}
```

]