Submitted by:

Faiyaz Ansari

ACKNOWLEDGMENT

**I would like to express my special thanks of gratitude to FLIP ROBO TECHNOLOGIES. as well as our SME Sajid Choudhary who gave me the opportunity to do this project on the topic Project Review &
Rating in which special thanks to our SME who helped to solve the problems .**

INTRODUCTION

Business Problem Framing

Here we need to predict ratings for the reviews which were written in the past and they don't have a rating. So, we have to build an application which can predict the rating by seeing the review.

Conceptual Background of the Domain Problem

We have a client who has a website where people write different reviews for technical products. Now they are adding a new feature to their website i.e. The reviewer will have to add stars(rating) as well with the review. The rating is out 5 stars and it only has 5 options available 1 star, 2 stars, 3 stars, 4 stars, 5 stars.

Motivation for the Problem Undertaken

The problem which I have faced while doing this project is that to scrap the data from Amazone websites and to scrap the laptops reviews and Ratings as well with huge no of 38400 rows data .am face many time only one problem that is finding the Rating

Model Building Phase

These steps I have followed to complete the project

1.Scraped the Reviews and Ratings from the amazon website and stored them into csv file .

2. Data Cleaning Exploratory

3. Data Pre-processing

4. Data Analysis

5. Model Building

6. Model Evaluation

7. Selecting the best model

Data Collection:

-For data collection I used Selenium library with python 3.6.
- First try to reach the main page of the Url(i.e: https://www.amazon.in/) then I try to heat the search button and entered the product then its go to that page and its collect all the product link and store it on a list then my driver is got the each link one by one ,when diver going to the inside of that product then its directly heat the rating tab ang it will gos to the another page ,once getting that page the driver trying to get the review from there and trying to click on that rating bar and getting the rating info then all this thinks are appending in a empty list .same are shown on the below pic .finally I got 38400 Data

In [7]:
```
def Amazon(amazon):
    driver.get(amazon)

    #search Bar
    search_path=driver.find_element_by_xpath("//div[@class='nav-search-field ']/input")
    search_path.send_keys(product_name)

    #search button
    search_button=driver.find_element_by_xpath("//input[@id='nav-search-submit-button']")
    search_button.click()
    for i in range(1,4):
        #getting links
        get_link=driver.find_elements_by_xpath("//div[@class='a-section a-spacing-medium']/div[2]/div[2]/div/div/h2/a")
        for i in get_link:
            Product_links.append(i.get_attribute('href'))
        try:
            nxt_btn=driver.find_element_by_xpath("//li[@class='a-last']/a")
            nxt_btn.click()
        except NoSuchElementException:
            pass

    for i in Product_links:
        driver.get(i)
        print("Review & Rating Scrapping ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶")
        #click in rating
        try:
            click_rating=driver.find_element_by_xpath("//a[1][@id='acrCustomerReviewLink']")
            click_rating.click()
        except NoSuchElementException or ElementClickInterceptedException:
            print("No rating avilable on this product ")
```

Data Cleaning Exploratory

Loading the DataSet

For loading the Data set I used below library below mention Picture

```
In [2]: import selenium
        import pandas as pd
        from selenium import webdriver
        from selenium.common.exceptions import NoSuchElementException,StaleElementReferenceException,ElementClickInterceptedException
        import time

        import warnings
        warnings.filterwarnings('ignore')
```

EDA

Next I thought to check the shape of my data set while I got it 38400 rows and 3 columns.
I drop the unwanted data (index columns) ,finally I have 38400 rows and 2 columns (product review and Product Rating).

```
In [15]: Review.head()
```

Out[15]:

	Product_Review	Product_Rating
0	1. Price is up a bit . In this price we are ge...	1.0
1	If it is come with pre loaded ms office nd 165...	1.0
2	The delivery was well on time. The product was...	5.0
3	Amazing product with proper delivery!!Perfect ...	5.0
4	Lenovo Legion 5i is a decent rig for gaming as...	4.0

Then I try to look on Nan values and find 200 Nan values are present in product review columns so I handel that ref _pic
I fill the nan values with " unknown",

```
In [6]: #lets check the null values
        df_amazon.isnull().sum()
```

```
Out[6]: Product_Review    200
        Product_Rating      0
        dtype: int64
```

here we can see that there is 200 null values present in the dataset

```
In [7]: #lets remove the null values
        df_amazon.dropna(inplace=True)
```

```
In [8]: #lets check it once again the null values to confirm
        df_amazon.isnull().sum()
```

```
Out[8]: Product_Review    0
        Product_Rating    0
        dtype: int64
```

Data Preprocessing:

```
In [1]: #importing Necessary Libraries
        import numpy as np
        import pandas as pd
        import seaborn as sns
        import matplotlib.pyplot as plt

        import warnings
        warnings.filterwarnings('ignore')
```

Then I used stopwords.

A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.

In [17]:
```
# Remove stopwords
import string
import nltk
from nltk.corpus import  stopwords

stop_words = set(stopwords.words('english') + ['u', 'ü', 'ur', '4', '2', 'im', 'dont', 'doin', 'ure'])

df_amazon['Product_Review'] = df_amazon['Product_Review'].apply(lambda x: ' '.join(
    term for term in x.split() if term not in stop_words))
```

Processing using lemmatization

```
from nltk.stem import SnowballStemmer, WordNetLemmatizer
stemmer = SnowballStemmer("english")
import gensim
def lemmatize_stemming(text):
    return stemmer.stem(WordNetLemmatizer().lemmatize(text,pos='v'))

#Tokenize and Lemmatize
def preprocess(text):
    result=[]
    for token in text:
        if len(token)>=3:
            result.append(lemmatize_stemming(token))

    return result
```

Result after punctuation after cleaning and removing punctuation.

```
    return result
```

```
In [20]: df_amazon.head()
```

Out[20]:

	Product_Review	Product_Rating	Review_len
0	buy product image quality average comparing sp...	1.0	203
1	let make one thing clear immediately xiaomi re...	4.0	2863
2	awesome	5.0	7
3	best buy price display quality amazing complai...	5.0	212
4	bad experience amazon bad camera quality phone...	1.0	228

Data Analysis:

Using the Word Cloud Library

```
l]: from wordcloud import WordCloud
#Getting sense of words in Rating 1
one = df_amazon['Product_Review'][df_amazon['Product_Rating']==1]

one_cloud = WordCloud(width=700,height=500,background_color='white',max_words=200).generate(' '.join(one))

plt.figure(figsize=(10,8),facecolor='r')
plt.imshow(one_cloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```

Feature Extraction:

Converting text to numeric used TfidfVectorizer.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split

tf_vec = TfidfVectorizer()
features = tf_vec.fit_transform(df_amazon['Product_Review'])

x = features
y = df_amazon['Product_Rating']
```

```
print("shape of x is :",x.shape)
print("shape of y is :",y.shape)
```

```
shape of x is : (29800, 1357)
shape of y is : (29800,)
```

Model Building

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression,PassiveAggressiveClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import AdaBoostClassifier

# Model selection libraries...
from sklearn.model_selection import cross_val_score, cross_val_predict, train_test_split
from sklearn.model_selection import GridSearchCV


# Importing some metrics we can use to evaluate our model performance....
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.metrics import roc_auc_score, roc_curve, auc
from sklearn.metrics import precision_score, recall_score, f1_score


# Creating instances for different Classifiers

LR=LogisticRegression()
MNB=MultinomialNB()
PAC=PassiveAggressiveClassifier()
DT=DecisionTreeClassifier()
Ad=AdaBoostClassifier()
RF=RandomForestClassifier()
```

```
def max_acc_score(clf,x,y):
    max_acc_score=0
    final_r_state=0
    for r_state in range(42,100):
        x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=.30,random_state=r_state,stratify=y)
        clf.fit(x_train,y_train)
        y_pred=clf.predict(x_test)
        acc_score=accuracy_score(y_test,y_pred)
        if acc_score > max_acc_score:
            max_acc_score=acc_score
            final_r_state=r_state
    print('Max Accuracy Score corresponding to Random State ', final_r_state, 'is:', max_acc_score)
    print('\n')
    return final_r_state
```

Interpreting the results for the above algorithms.

```
In [33]: score=pd.DataFrame({'Model':Model,'Learning Score':score,'Accuracy Score ':acc_score})
         score
```

Out[33]:

	Model	Learning Score	Accuracy Score
0	MultinomialNB	0.938974	0.943848
1	PassiveAggressiveClassifier	0.938974	0.943848
2	DecisionTreeClassifier	0.938974	0.943848
3	AdaBoostClassifier	0.938974	0.943848
4	RandomForestClassifier	0.938974	0.943848

After predicting the model we can find that we have got the all the model performed better in training dataset when it comes to testing data we can see that Random Forest and Decision Tree Classifier are performed better as compared to other models.

CONCLUSION

Key Findings and Conclusions of the Study

The key findings that I have find that I have scraped it from only one websites due to dead line I was able to scrap it .if I could scrap more websites we will get more better model prediction.

By using 38400 data we for two best models Random Forest Classifier and Decision Tree Classifier. Because of limited data I haven't go for sampling only just used stratify method to balance the data.

Limitations of this work and Scope for Future Work

In some algorithms where was taking to much time to execute but it was executed it in better way.
because of that laptops where getting hang and as we accept we got better score in every model.