FLIP ROBO

Review & Rating Project

Submitted by:

Ranjit Maity

ACKNOWLEDGMENT

I would like to express my special thanks of gratitude to FLIP ROBO TECHNOLOGIES. as well as our SME Swati Rustagi who gave me the golden opportunity to do this wonderful 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

```python
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 ")
            pass

        #clicking in see all reviews
        try:
            click_reviwes=driver.find_element_by_xpath("//a[@class='a-link-emphasis a-text-bold']")
            click_reviwes.click()
        except NoSuchElementException:
            pass

        #Start scraping the details
        Star_page=1
        End_page=200
        for page in range(Star_page,End_page+1):
            try:
                Reviews=driver.find_elements_by_xpath("//div[@class='a-row a-spacing-small review-data']/span/span")
                for R in Reviews:
                    Product_Review.append(R.text.replace('\n',''))
            except NoSuchElementException:
                Product_Review.append("Nan")
            try:
                Rating=driver.find_elements_by_xpath("//div[@class='a-section celwidget']/div[2]/a[1]")
                for i in Rating:
                    rat=i.get_attribute('title')

                    Product_Rating.append(rat[:3])
            except NoSuchElementException:
                Product_Rating.append("Nan")
            print("Product review and rating  of page {} scraped ".format(page+1))
            try:
                next_page=driver.find_element_by_xpath("//div[@id='cm_cr-pagination_bar']/ul/li[2]/a")
                if next_page.text=='Next→':
                    next_page.click()
                    time.sleep(4)
            except NoSuchElementException:
                pass
```

2. Data Cleaning Exploratory

2.1 Loading the DataSet

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

```
#Importing Necessary Library
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

After that I load the dataset with below command:

df_rating=pd.read_csv("D:/DATA_SCIENCE\I/intern Assignment/Review_rating/Review_Rating_final.csv")

2.2 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).

```
df_rating.head()
```

	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",

```
df_rating.isnull().sum()
```

```
Product_Review    200
Product_Rating      0
dtype: int64
```

```

```

```
df_rating.Product_Review.fillna("unknown",inplace=True)
df_rating.isnull().sum()
```

```
Product_Review    0
Product_Rating    0
dtype: int64
```

3.Data Preprocessing:

For Data Cleaning I used below library mention on picture

```
import nltk
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer, WordNetLemmatizer
import re
import string
```

Then I made a user define function to convert the all text into lower case

```
def Text_cleaning(df,col):
    column=df[col]
    Text=column.str.lower()
    column=Text
    return df
```

Similarly I made some more function as shown on below pic

```
def Remove_Punctuatuation(df,col):
    final=[]
    for i in range(0, len(df)):
            review = re.sub('[^a-zA-Z]', ' ', df[col][i])
            review = ' '.join(review)
            final_result=final.append(review)
    return final_result
```

On the above function am try to remove all the unwanted digit or symbol which are we know as Punctuation.

Next, I made one more function to remove the stop word .

What is Stop word?

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.

```
stop_words = set(stopwords.words('english') + ['u', 'ü', 'ur', '4', '2', 'im', 'dont', 'doin', 'ure'])
def Remove_Stop_Words(df,col):
    df[col]=df[col].apply(lambda x: ' '.join(term for term in x.split() if term not in stop_words))
    return df
```

Once I removed my Stop words then I go for the word

I word to Lemmatization my words ..

What is Lemmatization?

```
#steaming
lemmatizer = WordNetLemmatizer()
def stemming(col):
    lema=lemmatizer.lemmatize(col,pos='a')
    return lema
```

Then I made one more function where I passed stemming function and get the final result..

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

Lastly I make the last function and passed all function whatever I ede through that.

```
def data_cleaning(df,col):
    Text_cleaning(df,col)
    Remove_Punctuatuation(df,col)
    Remove_Stop_Words(df,col)
    stemming(col)
    preprocess(col)
    return df
```

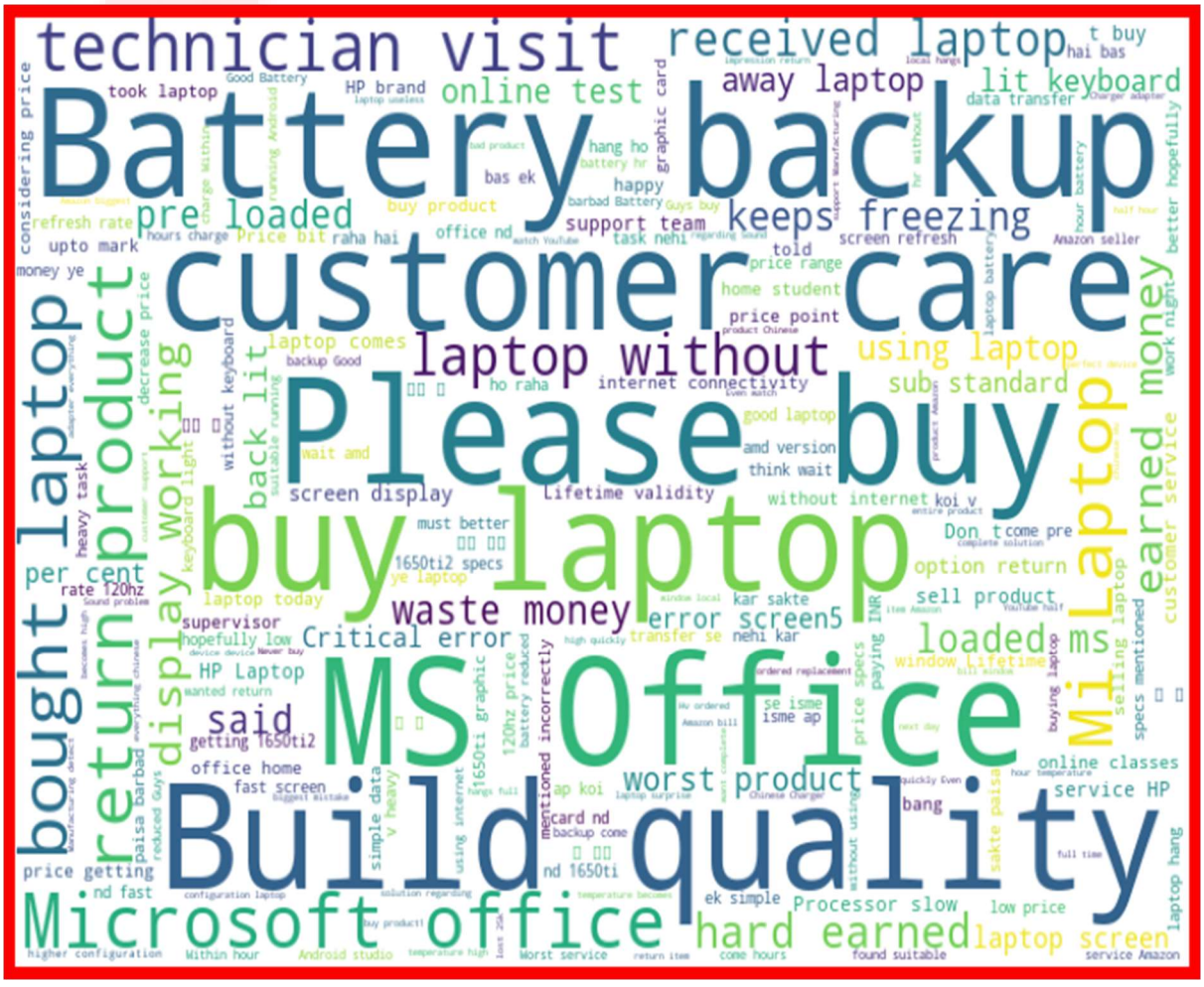
After run my final function I got the below result..

	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

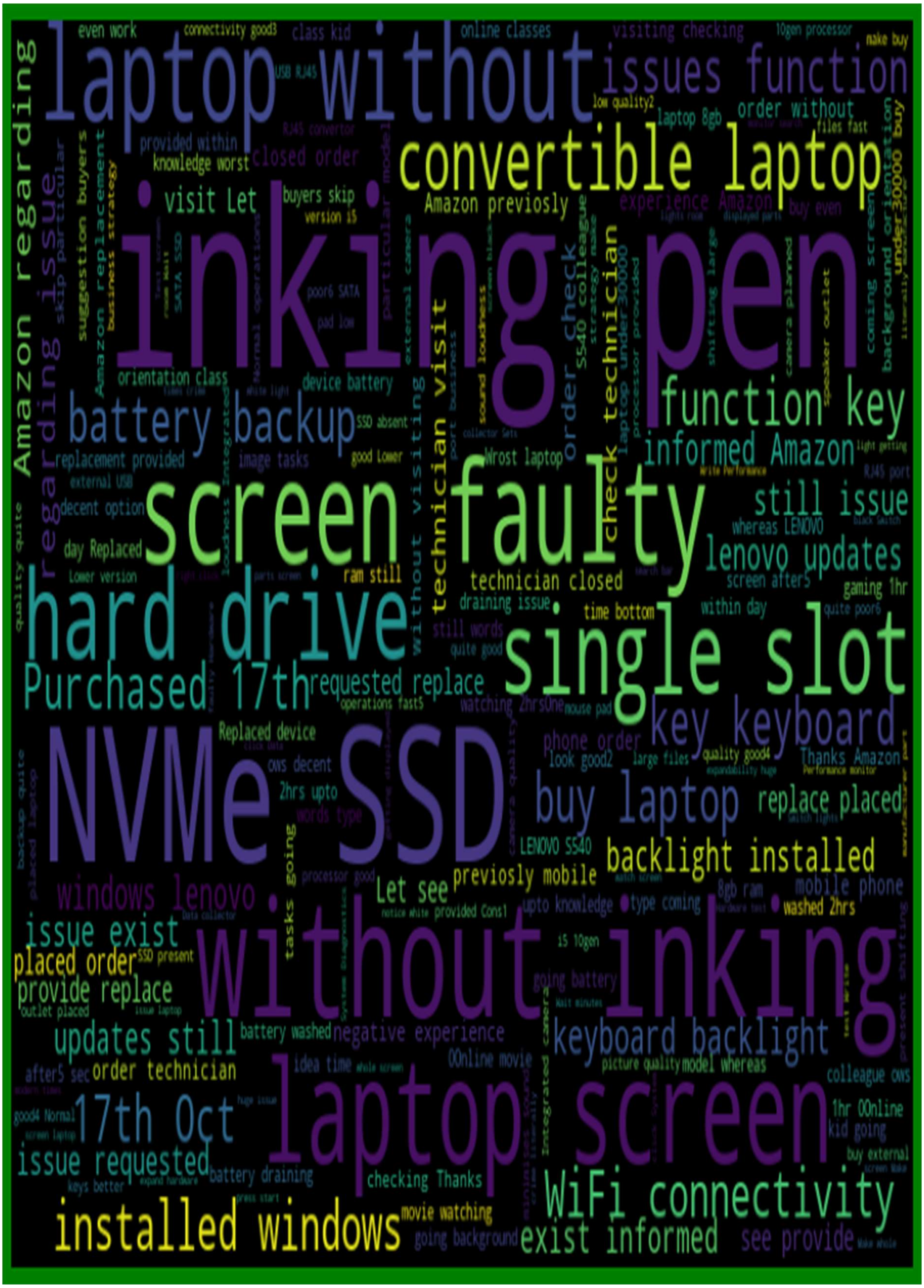
4.Data Analysis:

For Analysis purpose I used the Word Cloud Library

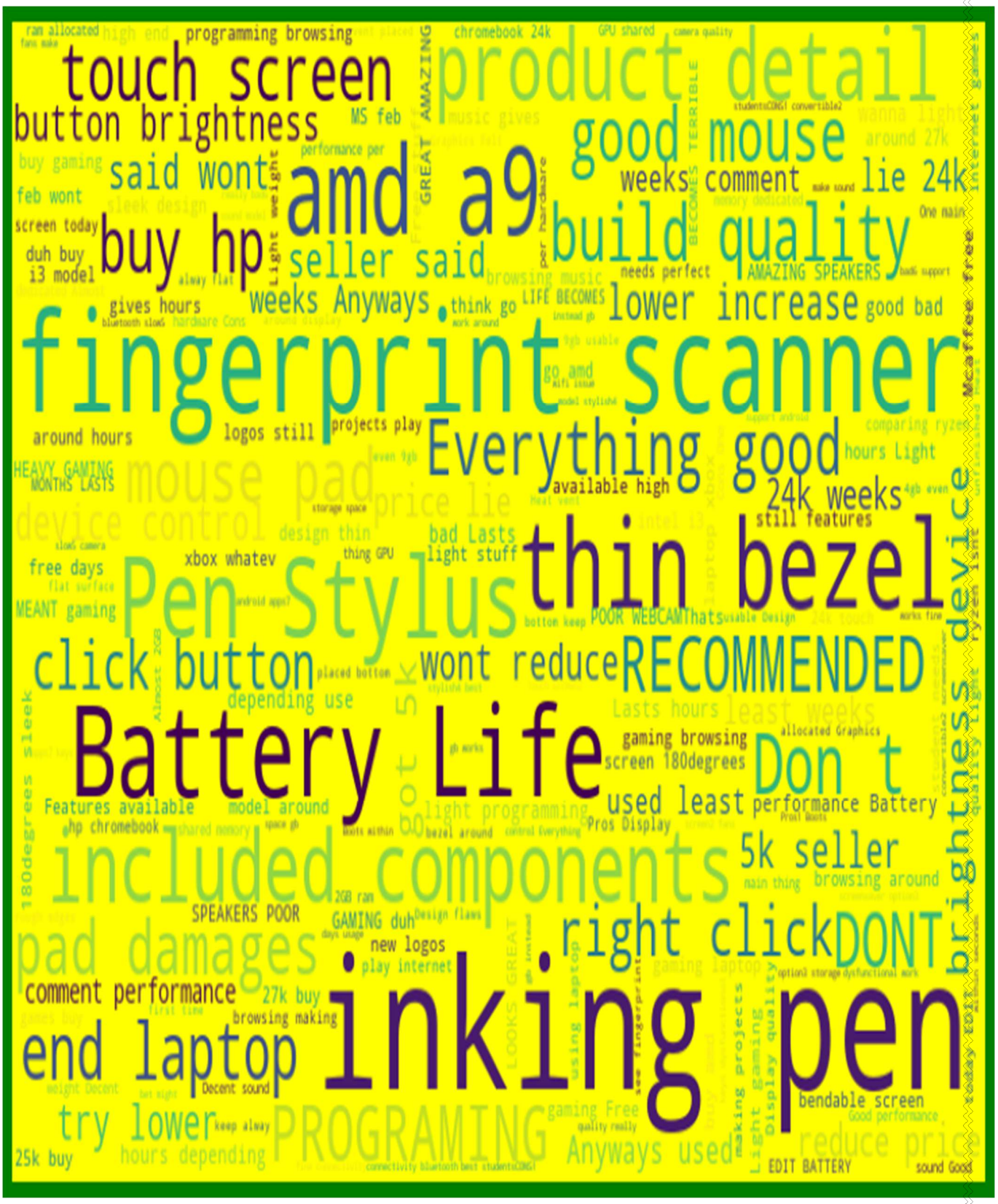
For 1 Rating Which words are making noise I shows that with cloud



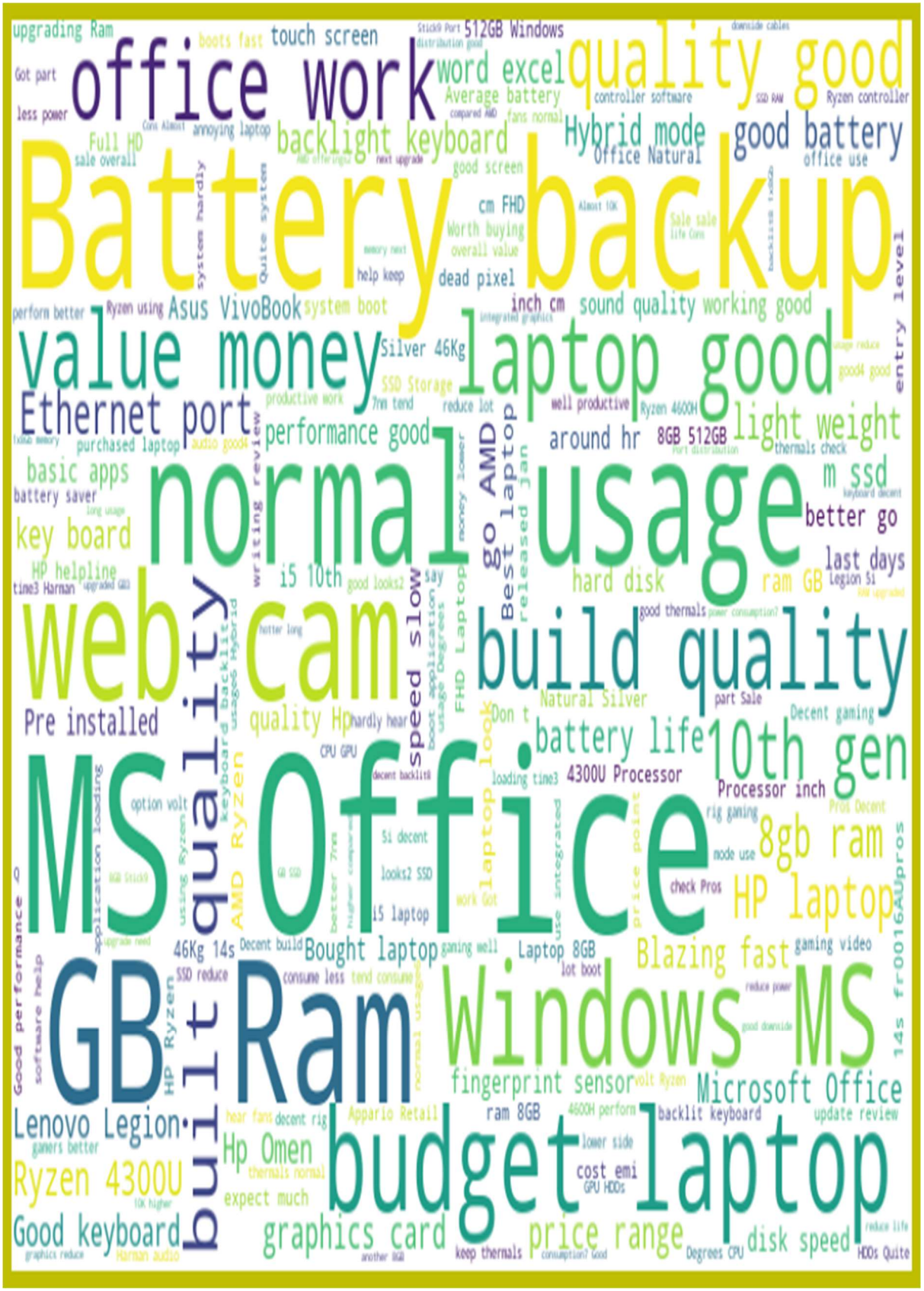
For 2 Rating Which words are making noise I shows that with cloud



For 3 Rating Which words are making noise I shows that with cloud



For 4 Rating Which words are making noise I shows that with cloud



For 5 Rating Which words are making noise I shows that with cloud

Feature Extraction:

For Convert the text to numeric I used TfidfVectorizer.

```
# 1. Convert text into vectors using TF-IDF

from sklearn.feature_extraction.text import TfidfVectorizer

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

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

5.Model Building

For model building I made a function to find the correct Random_state for getting the better Accuracy score ..

```
#Lets make a function for getting the best random_satae for a model toget better accuracy score
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.metrics import roc_auc_score, roc_curve, auc
final_random_state=[]
final_r2score=[]
model=[]
def max_acc(rgr,x,y):
    max_acc=0
    model.append(rgr)
    for r in range(42,100):
        x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=r,test_size=0.20)
        rgr.fit(x_train,y_train)
        y_prd=rgr.predict(x_test)
        rc=accuracy_score(y_test,y_prd)
        if rc>max_acc:
            max_acc=rc
            final_r=r
    final_random_state.append(final_r)
    final_r2score.append(max_acc)
    print("max accuracy_ score coressponding to ++>",final_r,"is++",max_acc*100)
```

Again I make one fore function for getting the Cross validation score

```
#lets make a function for cross_val_score
from sklearn.model_selection import cross_val_score
cvs=[]
def k(model,x,y):
    c=cross_val_score(model,x,y,cv=5,scoring="accuracy")
    print("mean accuracy score for ",model,c.mean())
    print("Standard deviation  in accuracy score for ",model,c.std())
    print()
    print("*****************************************************")
    print("After seen the cross validation score of",model,"the accuracy score mean is",c.mean())
    cvs.append(c.mean())
```

Lastly I Meade the very last function for ploting the Confusion matrix .

```
def PLT(md,x,y,rd):
    x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=rd,test_size=0.20)
    md.fit(x_train,y_train)
    pre=md.predict(x_test)
    acc=accuracy_score(y_test,pre)
    print(acc*100)
    cm=confusion_matrix(y_test,pre)
    print()
    print()
    sns.heatmap(cm,annot=True)
    plt.show()
    cr=classification_report(y_test,pre)
    print()
    print()
    print()
    print(cr,"\n","ooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo")
```

Next I used below mentions algorithms

Logistic Regression()

MultinomialNB()

PassiveAggressiveClassifier()

DecisionTreeClassifier()

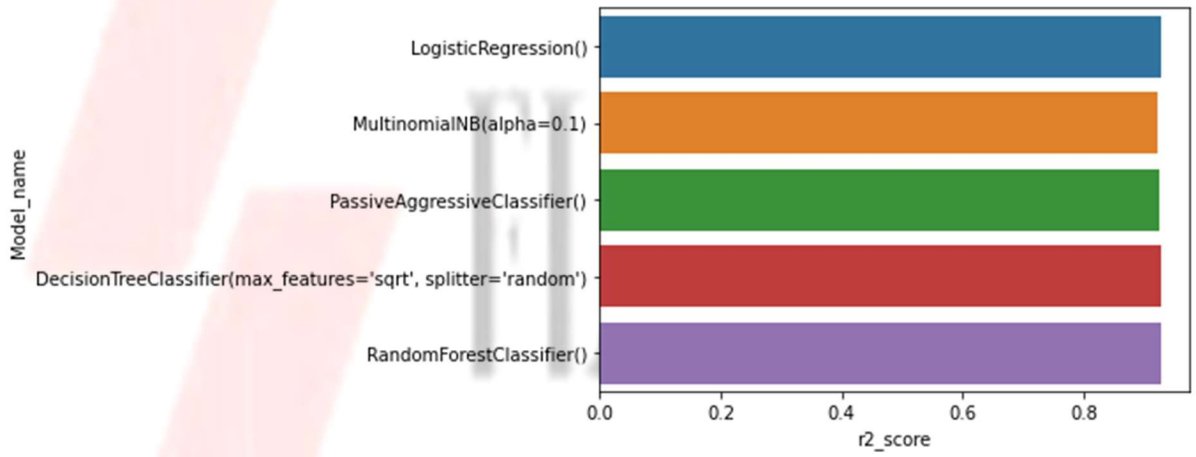
AdaBoostClassifier()

RandomForestClassifier()

Xgboost Classifier()

After Used all the Algorithms I got the Result as

	Model_name	r2_score	Random_State
0	LogisticRegression()	0.927604	73
1	MultinomialNB(alpha=0.1)	0.922396	43
2	PassiveAggressiveClassifier()	0.925260	81
3	DecisionTreeClassifier(max_features='sqrt', sp...	0.927604	73
4	(DecisionTreeClassifier(max_features='auto', r...	0.927604	73



Finally am saving the Model as RandomforestClassifier.

Interpretation of the Results

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.