NAME OF THE PROJECT

RATINGS PREDICTION PROJECT

Submitted by:

Muktikanta Sahoo

ACKNOWLEDGMENT

I would like to express my deep and sincere gratitude to my Project supervisor, Miss. Sapna Verma, Project coordinator, Flip Robo Technologies for giving me the opportunity to do research and providing invaluable guidance throughout this project. His dynamism, vision, sincerity and motivation have deeply inspired me.

Finally, my thanks go to all the people who have supported me to complete the research work directly or indirectly.

INTRODUCTION

- ***Business Problem Framing***
  When you go to make an online purchase, what's the first thing you do? In an ecommerce-driven world where customers can't physically experience products before purchasing, many consumers turn to online product reviews.
  As online review sites such as Yelp! and Facebook have expanded, finding an opinion on just about anything is only a few clicks away. The proliferation of reviews has even gone so far as to shape how businesses are perceived online
  For any company that exists in the digital space, online reviews are critically important when it comes to winning business and maintaining a positive reputation
- ***Conceptual Background of the Domain Problem***
  In today's web-based world, virtually everyone is reading online reviews. In fact, 99% of people read them and 60% trust them as much as they would a personal recommendation. The effects of reviews are measurable, too.
  Negative reviews can carry as much weight as positive ones. One study found that 82% of those who read online reviews specifically seek out negative reviews..
  That may sound alarming — this stat only emphasizes that negative reviews aren't going unnoticed — but there are some benefits: Research indicates that users spend five times as long on sites when interacting with negative reviews, with an 85% increase in conversion rate.
- ***Review of Literature***
  Reviews are able to garner trust because they represent personal and unbiased users' experiences. Written by customers, they constitute unique content that search engines love. Naturally, a good number of Customer reviews will drive higher organic traffic to Ecommerce websites. There is a great demand for original content that equally impresses the customers and search engines. User-generated content is authentic and distinctive. Since it is created by customers, brands need not use much of their resources to build it. User-Generated Content is the solution to the ongoing search for original content. It is a steady stream of quality, searchable content for a brand's website. User-Generated Content also helps brands learn about the latest trends and the preferences of their customers..
- ***Motivation for the Problem Undertaken***
  Let us imagine that you've purchase one mobile (I Phone-13 Pro Max What would you search for, 'Iphone-13, 'flip kart or Amazon' you finalised the last option, you belong to the majority of people who use more than four words in their search query. These are called long-tail keywords.

Analytical Problem Framing

- ***Mathematical/ Analytical Modelling of the Problem.***

  In our scrapped dataset, our target variable "Rating " is a categorical variable i.e., it can be classified as '1.0', '2.0','3.0','4.0','5.0'. Therefore, we will be handling this modelling problem as classification

  <u>This project is done in two parts</u>
  1- Data Collection Phase
  2- Model Building Phase

- ***Data Sources and their formats***
  We have an E-Commerce Site(Flip-kart) 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. Now they want 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.
- ***Data Pre-processing Done***
  Using NLTK library have followed multiple technique for data pre-processing.

```
#Converting all messages to lowercase
df[df_column_name] = df[df_column_name].str.lower()

#Replace email addresses with 'email'
df[df_column_name] = df[df_column_name].str.replace(r'^.+@[^\.].*\.[a-z]{2,}$','emailaddress')

#Replace URLs with 'webaddress'
df[df_column_name] = df[df_column_name].str.replace(r'^http\://[a-zA-Z0-9\-\.]+\.[a-zA-Z]{2,3}(/\S*)?$','webaddress')

#Replace money symbols with 'dollars' (£ can by typed with ALT key + 156)
df[df_column_name] = df[df_column_name].str.replace(r'£|\$', 'dollars')

#Replace 10 digit phone numbers (formats include paranthesis, spaces, no spaces, dashes) with 'phonenumber'
df[df_column_name] = df[df_column_name].str.replace(r'^\(?[\d]{3}\)?[\s-]?[\d]{3}[\s-]?[\d]{4}$','phonenumber')

#Replace numbers with 'numbr'
df[df_column_name] = df[df_column_name].str.replace(r'\d+(\.\d+)?', 'numbr')

#Remove punctuation
df[df_column_name] = df[df_column_name].str.replace(r'[^\w\d\s]', ' ')

#Replace whitespace between terms with a single space
df[df_column_name] = df[df_column_name].str.replace(r'\s+', ' ')

#Remove leading and trailing whitespace
df[df_column_name] = df[df_column_name].str.replace(r'^\s+|\s+?$', '')

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

```
#Calling the class
clean_text(df, 'Review')
df['Review'].tail(3)
```

```
25717    camera good beginner getting much high quality...
25718    product good build quality also special thanks...
25719    verry nice opinion great camera price range lo...
Name: Review, dtype: object
```

```
#Tokenizing the data using RegexpTokenizer
from nltk.tokenize import RegexpTokenizer
tokenizer=RegexpTokenizer(r'\w+')
df['Review'] = df['Review'].apply(lambda x: tokenizer.tokenize(x.lower()))
df.head()
```

```
#Processing review with above Function
processed_review = []

for doc in df.Review:
    processed_review.append(preprocess(doc))

print(len(processed_review))
processed_review
```

```
    'use',
    'io',
    'devic',
    'android',
    'user',
    'past',
    'numbr',
    'year',
    'order',
    'iphon',
    'numbr',
    'numbrgb',
    'product',
    'red',
    'experi',
    'use',
    'numbr',
    'week',
    'numbr',
    'deliveri'],
```

```
#Assigning this to the dataframe
df['clean_review']=processed_review
df.head()
```

```
df['Review'] = df['clean_review'].apply(lambda x:' '.join(y for y in x))
df.head()
```

	Rating	Review	clean_review
0	5	realli satisfi product receiv total genuin pac...	[realli, satisfi, product, receiv, total, genu...
1	5	great iphon snappi experi appl kind upgrad iph...	[great, iphon, snappi, experi, appl, kind, upg...
2	5	amaz phone great camera better batteri give be...	[amaz, phone, great, camera, better, batteri, ...
3	5	first io phone happi product much satisfi love...	[first, io, phone, happi, product, much, satis...
4	5	previous use one plus numbrt great phone decid...	[previous, use, one, plus, numbrt, great, phon...

- ***Hardware and Software Requirements and Tools Used***

  1. Python 3.8.
  2. NumPy.
  3. Pandas.
  4. Matplotlib.
  5. Seaborn.
  6. Data science.
  7. SciPy
  8. Sklearn.
  9. NLTK library.
  10. Machine learning.
  11. CPU with RAM of 8GB.
  12. Anaconda Environment.
  13. Jupyter Notebook.
  14. re (Regular expression

Model/s Development and Evaluation

- **Listing down all the algorithms used for the training and testing**

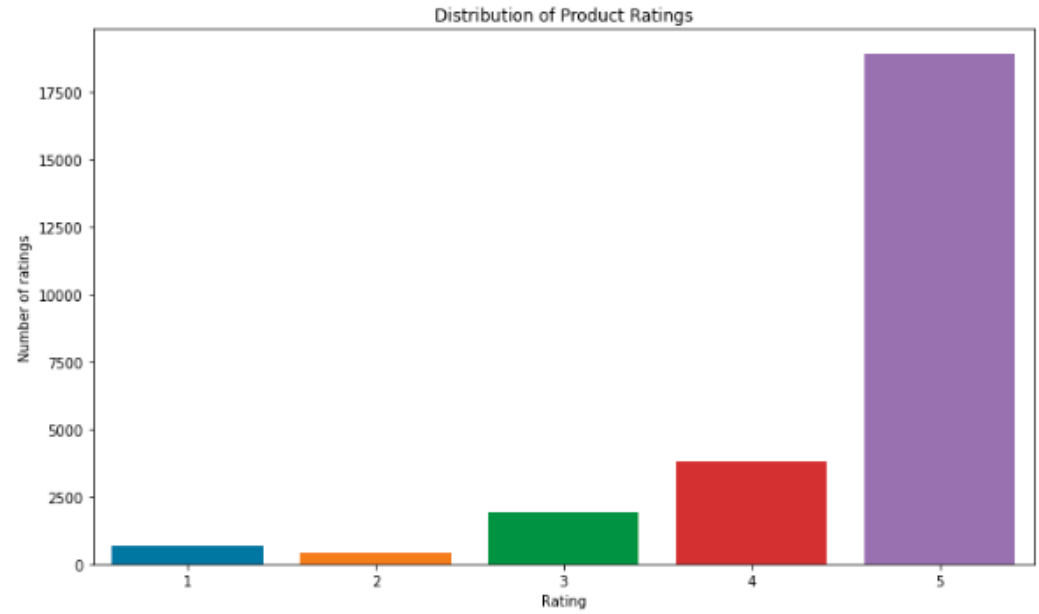
```
#Initializing the instance of the model
LR=LogisticRegression()
mnb=MultinomialNB()
dtc=DecisionTreeClassifier()
knc=KNeighborsClassifier()
rfc=RandomForestClassifier()
abc=AdaBoostClassifier()
gbc=GradientBoostingClassifier()
```

```
models= []
models.append(('Logistic Regression',LR))
models.append(('MultinomialNB',mnb))
models.append(('DecisionTreeClassifier',dtc))
models.append(('KNeighborsClassifier',knc))
models.append(('RandomForestClassifier',rfc))
models.append(('AdaBoostClassifier',abc))
models.append(('GradientBoostingClassifier',gbc))
```

Run and Evaluate selected mod

```
#making a for loop and calling the algorithm one by one and save data to respective model using append function
Model=[]
score=[]
cvs=[]
rocscore=[]
for name,model in models:
    print('*******************************',name,'**************************')
    print('\n')
    Model.append(name)
    model.fit(x_train,y_train)
    print(model)
    pre=model.predict(x_test)
    print('\n')
    AS=accuracy_score(y_test,pre)
    print('accuracy_score: ',AS)
    score.append(AS*100)
    print('\n')
    sc=cross_val_score(model,x,y,cv=5,scoring='accuracy').mean()
    print('cross_val_score: ',sc)
    cvs.append(sc*100)
    print('\n')
    print('Classification report:\n ')
    print(classification_report(y_test,pre))
    print('\n')
```

- ***Visualizations***



CONCLUSION

- ***Key Findings and Conclusions of the Study***

  Online reviews are an effective word of mouth marketing strategy in the digital age, providing outside perspectives on products and services. While positive reviews can drive revenue and build a trustworthy reputation, negative reviews or the absence of reviews can do the opposite. Understanding the importance of reviews as well as how to leverage them to boost your business can be a critical way to get ahead in the competitive ecommerce marketplace, positioning yourself miles ahead of the competition..

- ***Learning Outcomes of the Study in respect of Data Science***

  In this study we understand how the reviews plays major role in the business of E-Commerce. On a lay perspective how, ratings play a major role in promoting the business and most of the people are more interested in only knowing what is the rating of the products rather than the reviews.

- ***Limitations of this work and Scope for Future Work***

  Generating unique, high-quality content on every product page is a big challenge for Ecommerce businesses. Scaling content for thousands of products in the inventory is a Herculean task that online retailers cannot possibly handle on their own as it demands a lot of time and resources. To add to the complexity of search engine ranking, Google algorithms are becoming more intuitive with every update. They filter out websites that do not have original content and those with duplicate content within the site. But on the bright side, when the content on a website is unique and valuable, Google recognizes it for being trustworthy. So, if a brand wants to stand out in search results, it is not enough to just generate unique content for the website, but also to ensure that the unique content is compelling and valuable.