FLIP ROBO

RATINGS PREDICTION PROJECT

Submitted by:

Abhimanyu Varpe

ACKNOWLEDGMENT

I would like to express my special thanks to my mentor Miss Sapna Verma mam who gave me her support and assistance throughout the project. I am thankful to Fliprobo Technologies and Data Trained Institute to give me guidelines and knowledge to complete this project.

Links and websites that I preferred:

https://towardsdatascience.com/oversampling-and-undersampling-5e2bbaf56dcf
https://career-resource-center.udacity.com/portfolio/data-sciencereports#:~:text=A%20data%20science%20report%20is,the%20legitimacy%20o f%20your%20process

INTRODUCTION

- ## Business Problem Framing

  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. 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.

- ## Conceptual Background of the Domain Problem

  One who has knowledge of data collection it will be a great help for them because more the data is accurate and helpful better model we will get.
  Also now a days it is very common to shop from e commerce websites so it helpful to scrap the data that which data to take and what not.
  But for this problem only reviews is important.

- ## Review of Literature

  It is helpful for those who have experience of e-shopping and on which basis they use to order the products. It will help to work on data having more than 20k rows in the dataset the domain knowledge and experience can help to understand the correlated things and changing trends over the period of time.

- Motivation for the Problem Undertaken

Shopping from e commerce is common now a days, I order to have the trusted customers and delivering required service reviews and ratings is most important factor for any e commerce website.

This problem will be helpful for e commerce websites and customers for future business and shopping.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

I have collected the data through web scraping from different e-commerce websites. Websites from where I have scraped data is amazon, flipkart, paytm mall, snapdeal and digit. Products I have scraped are Mobile phones, Home theatre, Smart watch, camera, Monitor, Headphones, printer etc. Dataset contains 20k plus rows and four columns.

- Data Sources and their formats

As I have scraped the data from different websites it has variety of data and also this data contains some duplicates values or not required values because as on the web pages of the e commerce websites some products got repeated. Below is the snapshot of reading the dataset and printing top five rows of the dataset.

```
import seaborn as sns
```

```
df = pd.read_csv('ppf2.csv')
```

```
df.head() #change this
```

	reviews	Price	Brand	Ratings
0	boss of all bluetooth speakerboommmmmm box tha...	₹19,999	Home Theatre	5
1	amazing speaker with superb bass it comes with...	₹6,499	Yes	5
2	excellent product in budget i m loving iti pre...	₹1,999	Home Theatre	5
3	sound quality is good and loud but you cant ge...	₹2,799	Home Theatre	5
4	the best product in low price osm the sound qu...	₹2,499	No	5

- Data Preprocessing Done

  Pre-processing Pipe Line is an important step towards the data modelling, to make data ready for prediction. Following Steps I followed for Pre Processing:

  As In this case for the reviews column I have performed various techniques to clean the text data following are the techniques I have used:

  - **Word Tokenize**: Split the reviews into the words
  - **Removal of stop words** : This technique decreased the length of thereview
  - **Word lemmatize** : Fetching the root word for the spelling
  - **Remove punctuation** : Remove punctuation from the reviews
  - **Remove Emoji's and numbers:** Remove emoji and number as that will create some problem in vectorization.
  - **Treating the ratings column** : Filling the null values and round of it values
  - **Encoding:** Label encoding for the ratings column.

```
In [5]: df['reviews'] = df['reviews'].str.lower()

In [6]: df['reviews'] = df['reviews'].str.translate(str.maketrans(' ',' ',string.punctuation))

In [11]: stop_words = set(stopwords.words('english'))

         df['review'] = df['review'].apply(lambda x:[term for term in x if term not in stop_words])

In [29]: lemmatizer = WordNetLemmatizer()
         def word_lemmatizer(text):
             lem = [lemmatizer.lemmatize(i) for i in text]
             return lem

In [30]: df['review'] = df['review'].apply(lambda x: word_lemmatizer(x))

In [9]: df['review'] = df['reviews'].apply(nltk.word_tokenize)
```

```
In [61]: from sklearn.preprocessing import LabelEncoder
         lab_en = LabelEncoder()
         df['Ratings'] =lab_en.fit_transform(df['Ratings'])
```

- ## Data Inputs- Logic- Output Relationships

  Here the two columns price and brand will not be useful to predict the ratings I will be using Reviews only to predict the data.

  So purely the output variable has five categories kind of multi class classification and model will learn to predict the ratings based on the words used in the reviews.

  Output variable is unbalanced so I will be using Oversampling for that.

- ## Hardware and Software Requirements and Tools Used

  Tool: Jupyter Notebook 6.1.4
  • Web-based interactive computing notebook Environment.
  Software Requirement:
  • The client environment may be Windows, macOS, or Linux.
  Hardware Requirement:
  • CPU: 2 x 64-bit, 2.8 GHz, 8.00 GT/s CPUs or better.
  • **Memory:**
  Minimum RAM size of 32 GB, or 16 GB RAM with 1600 MHz DDR3 installed, for a typical installation with 50 regular users.
  Libraries:
  • Pandas: For reading CSV file, Converting dataset into a data frame, handling date data type, and more.
  • Seaborn and matplotlib: For EDA and Visualization , nltk and gensim

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  For Solving this problem the most important and time consuming task is to perform preprocessing on the text data like remove emojis , numbers , single character and spelling correction .

  And to make it simple I have rounded off the ratings and encoded it through label encoder which will basically assign the prediction from 1 to 5 star only.

  Also rather than treating it as a categorical problem we can go for the regression algorithms also because the values of ratings lies from 1.0 to 5.0

  This could be the solution.

  But best practice is to do that solution.

- Testing of Identified Approaches (Algorithms)

  Before do modelling first I did Oversampling because unbalanced data will not give the suitable model or prediction.
  Algorithms I have used:

  1. Multinomial Algorithm :
     Accuracy score: 0.72 F1 Score:

  2. SVC:

     Accuracy score: 0.88 F1 Score: 0.91

3. Decision Tree Classifier:
   Accuracy score: 0.87 F1 Score:0.91

4. KNeighborsClassifier:
   Accuracy score: 0.82 F1 Score: 0.89

Run and Evaluate selected modelsDifferent Algorithms Results:

1. Multinomial Algorithm:

```
In [82]: X_train,x_test,Y_train,y_test = train_test_split(x1,y1,random_state=44)

In [83]: naive.fit(X_train,Y_train)

         y_pred= naive.predict(x_test)

         print(accuracy_score(y_test,y_pred))

         0.720775241305347
```

2. SVC:

```
In [88]: ob1  =  SVC()
         ob1.fit(X_train,Y_train)
         y_pred= ob1.predict(x_test)

         print(accuracy_score(y_test,y_pred))

         0.8839436188141566

In [93]: from sklearn import metrics

In [95]: print(classification_report(y_pred,y_test))
         conf_mat = confusion_matrix(y_test,y_pred)

         ax=plt.subplot()

         sns.heatmap(conf_mat,annot=True)
```

	precision	recall	f1-score	support
0	0.85	0.99	0.91	2211
1	0.89	1.00	0.94	2277
2	0.86	0.96	0.91	2389
3	0.87	0.71	0.79	3184
4	0.95	0.83	0.88	2993
accuracy			0.88	13054
macro avg	0.88	0.90	0.89	13054
weighted avg	0.89	0.88	0.88	13054

3. Decision Tree Classifier:

In [98]:
```
ob2  =  DecisionTreeClassifier()
ob2.fit(X_train,Y_train)
y_pred_ = ob2.predict(x_test)

print(accuracy_score(y_test,y_pred_))
```

```
0.8774322046882181
```

In [99]:
```
print(classification_report(y_pred_,y_test))
conf_mat = confusion_matrix(y_test,y_pred_)

ax=plt.subplot()

sns.heatmap(conf_mat,annot=True)
```

```
              precision    recall  f1-score   support

           0       0.85      0.99      0.91      2231
           1       0.89      1.00      0.94      2283
           2       0.87      0.92      0.89      2528
           3       0.89      0.70      0.78      3278
           4       0.89      0.86      0.87      2734

    accuracy                           0.88     13054
   macro avg       0.88      0.89      0.88     13054
weighted avg       0.88      0.88      0.87     13054
```

4. KNeighborsClassifier:

In [101]:
```
ob3  =  KNeighborsClassifier()
ob3.fit(X_train,Y_train)
y_pred_k = ob3.predict(x_test)

print(accuracy_score(y_test,y_pred_k))
```

```
0.82020836525203
```

In [102]:
```
print(classification_report(y_pred_k,y_test))
conf_mat = confusion_matrix(y_test,y_pred_k)

ax=plt.subplot()

sns.heatmap(conf_mat,annot=True)
```

```
              precision    recall  f1-score   support

           0       0.85      0.94      0.89      2343
           1       0.91      0.95      0.93      2441
           2       0.85      0.92      0.88      2493
           3       0.89      0.59      0.71      3913
           4       0.61      0.86      0.71      1864

    accuracy                           0.82     13054
   macro avg       0.82      0.85      0.82     13054
weighted avg       0.84      0.82      0.82     13054
```

- ## Key Metrics for success in solving problem under consideration

  For selecting the best model I have used accuracy score and classification report and confusion matrix.

  In classification problems we cannot rely only on the

  Accuracy score may be it shows the over fitting of the model.

  So In classification report we should look at the F1 score and errors in the confusion matrix.

  On basis of this three SVC was performing best.

  So I have done hyper parametric tuning of that model.

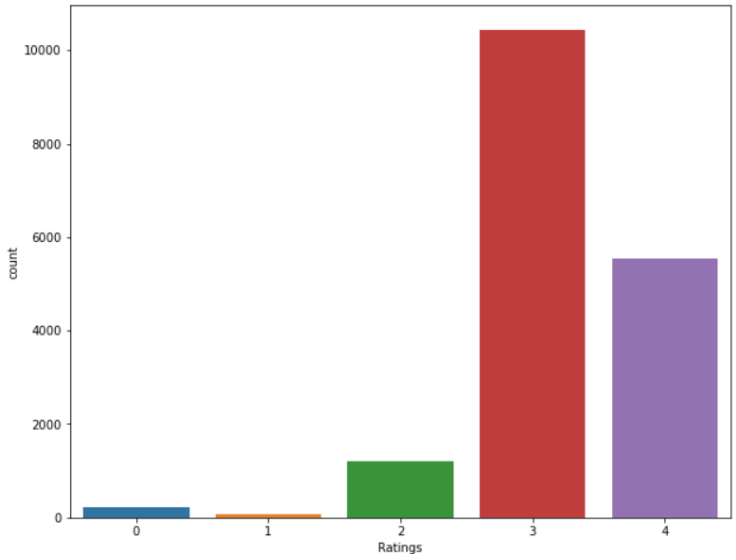
- ## Visualizations

  This Problem Does not require Visualization so far because of only onefeature and one output.
  But below are the plots I have plotted:

  ### 1. Countplot of Ratings:

```
In [64]: import matplotlib.pyplot as plt
         plt.figure(figsize=(10,8))
         sns.countplot(df['Ratings'])
```

```
Out[64]: <AxesSubplot:xlabel='Ratings', ylabel='count'>
```



2. Word cloud:

- Interpretation of the Results

As the ratings are unbalanced column, most frequent is 4 star.

May be reason being that I have collected data from most trustedwebsites.

Most occurring words are verified, purchase, good and helpful.

As SVC was performing best I have done hyper parametric tuning

where rbf is the best performing kernel.

CONCLUSION

- Key Findings and Conclusions of the Study

- Data collection is the most important and time consuming task in this problem, the more accurate and clean the data it more accurate it will give the prediction.

- The more variety and large data size more the result will be precise.

- For data collection different websites I have used to get diversified data.

- The problem will help to work on both the technologies like Natural language processing and machine learning.

- Preprocessing of review column is also a time consuming task and important task to be done.

- Learning Outcomes of the Study in respect of Data Science

  As I have scraped the data by my own it will be a good learning to know how to fetch the data and what data will be useful for which output.

  Secondly to learn how to use natural language processing to model with machine learning.

  And how to work one multiclass classification.

  Data size is large that will also give a point how to work on that.

- Limitations of this work and Scope for Future Work

  The data that I have collection is basically need more cleaning according to me, also here some websites have same products displayingmany time because of which same reviews we have to process that things we should work on.