RATINGS PREDICTION PROJECT

Submitted by:

Dolypona Das

ACKNOWLEDGMENT

I would like to thank our SME(Shubham Yadav) for his expert advice and encouragement throughout this project, and I also took some help from googled documents.

INTRODUCTION

- ## Business Problem Framing

- In this project, 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 of 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

1.Firstly we will check whether the problem is supervised or not, mostly we will have a supervised one where there will be a target variable.
2. After that we will check whether the project is a regression type or a classification type.
3. We will also check whether our dataset is balanced or imbalanced. If it is an imbalanced one, we will apply sampling techniques to balance the dataset.
4. Then we will do model building and check its accuracy.
5. Our main motto is to build a model with good accuracy and for that we will also go for hyperparameter tuning.

- ## Review of Literature

I am summarizing my research done on the topic.

  - I have collected my data using web scraping and make my dataset.
  - I have imported important libraries for my NLP project.

- I have created the dataframe for the train dataset. I have analysed my data by checking its shape, number of columns, presence of null values if any and checking the datatypes.
- Then I have done some data preprocessing steps, e.g. converting the train data into lowercase,removing punctuations and stopwords, checking the length of the clean data, converting text into vectors using TF-IDF, and splitting the data into independent and dependant variables .
- I have used Multinomial Naïve Bayes classifier for model building.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  If we look at data science, we are actually using mathematical models to model (and hopefully through the model to explain some of the things that we have seen) business circumstances, environment etc and through these models, we can get more insights

  such as the outcomes of our decision undertaken, what should we do next or how shall we do it to improve the odds. So mathematical models are important, selecting the right one to answer the business question can bring tremendous value to the organization. Here, I am using Multinomial Naïve Bayes classifier for model building.

- Data Preprocessing Done
  - I have dropped the column 'Index' as there is no use of it.
  - Then I have converted the train data into lowercase as it is an NLP project.
  - Then I have removed punctuations ,stopwords to get a clean length.
  - Then I have converted text into vectors using TF-IDF so that the data gets ready for model building.

- I have splitted the independent and dependant variables into x and y.

- Hardware and Software Requirements and Tools Used

Hardware requirements:

Processor: Intel(R) Celeron(R) CPU N3050 @1.60 GHz 1.60 GHz

RAM: 3.92 GB

System type: 64-bit operating system, x64-based processor

Software requirements:

Python: One of the most used programming languages

Tools used:

Jupyter notebook: Jupyter is a free, open-source,interactive web tool known as a computational notebook where I have written my python codes.

NumPy: NumPy is an open-source numerical Python library. NumPy contains a multi-dimensional array and matrix data structures.

Pandas: Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays.

Matplotlib: It is a Python library used for plotting graphs with the help of other libraries like Numpy and Pandas. It is a powerful tool for visualizing data in Python.

Seaborn: It is also a Python library used for plotting graphs with the help of Matplotlib, Pandas, and Numpy.

Scikit-learn: It is probably the most useful library for machine learning in Python.The sklearn library contains a lot of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction.

Scipy.stats: This module contains a large number of probability distributions as well as a growing library of statistical functions.

Natural Language Toolkit: NLTK is a leading platform for building Python programs to work with human language data. It provides

easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)
  - For training and testing the data, I have imported **train_test_split library** from scikit-learn.
  - For NLP model building, I have used **Multinomial NB classifier** on my train dataset .
- Testing of Identified Approaches (Algorithms)

  I have used Multinomial Naïve Bayes classifier for model building because generally in NLP projects, Multinomial NB is mostly used.

  Multinomial NB: Multinomial Naive Bayes algorithm is a probabilistic learning method that is mostly used in Natural Language Processing (NLP). The algorithm is based on the Bayes theorem and predicts the tag of a text such as a piece of email or newspaper article.
- Run and Evaluate selected models

In [32]:
```
#Convert text into vectors using TF-IDF
#Instantiate MultinomialNB classifier
#Split feature and Ratings

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report

tf_vec=TfidfVectorizer()
naive=MultinomialNB()
features=tf_vec.fit_transform(df['Reviews'])
X=features
y=df['Ratings']
```

In [33]:
```
#Train and predict
X_train,x_test,Y_train,y_test=train_test_split(X,y,random_state=42,)
naive.fit(X_train,Y_train)
y_pred=naive.predict(x_test)
print('Final score=>',accuracy_score(y_test,y_pred))
```

```
Final score=> 0.5984990619136961
```

- Key Metrics for success in solving problem under consideration

  1.Accuracy_score

  2.Confusion_matrix

  3.Classification_report

In [34]:
```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           1       0.00      0.00      0.00        26
           2       0.67      0.15      0.25        53
           3       1.00      0.07      0.13        71
           4       0.55      0.28      0.37        92
           5       0.60      0.96      0.74       291

    accuracy                           0.60       533
   macro avg       0.56      0.29      0.30       533
weighted avg       0.62      0.60      0.51       533
```

In [35]:
```
#plot confusion matrix heatmap
conf_mat=confusion_matrix(y_test,y_pred)
conf_mat
```

Out[35]:
```
array([[  0,   2,   0,   0,  24],
       [  0,   8,   0,   1,  44],
       [  0,   1,   5,   9,  56],
       [  0,   1,   0,  26,  65],
       [  0,   0,   0,  11, 280]], dtype=int64)
```

- Interpretation of the Results
  - In the preprocessing part ,I have cleaned my data in various ways like converting the text into lowercase, then removing punctuation, stopwords and finally converting the text into vectors using TF-IDF.
  - In the modelling part,I have designed the model using Multinomial NB where I have calculated accuracy score, classification report and confusion matrix.

CONCLUSION

- Key Findings and Conclusions of the Study

  The key findings are we have to study the data very clearly so that we are able to decide which data are relevant for our findings.

  The conclusion of our study is we have to achieve a model with good accuracy.
- Learning Outcomes of the Study in respect of Data Science

  We will develop relevant programming abilities. We will demonstrate proficiency with statistical analysis of data. We will develop the ability to build and assess data-based models. We will execute statistical analyses with professional statistical software. The best algorithm for this project according to my work is Multinomial Naïve Bayes classifier.
- Limitations of this work and Scope for Future Work

  To make my model more accurate and useful,I have to collect more data so that I can overcome underfitting and overfitting issues.