Ratings Prediction Project

Submitted by:

Manish Kr.

ACKNOWLEDGMENT

I would like to thank Keshav Bansal Sir, who helped me in this project by clearing my doubts in doubt clearing sessionsand providing me with learning material that proved to be helpful for my project. Also, DataTrained live sessions and Krish Naik youtube channels helped me understand concepts that were alien to me.

INTRODUCTION

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.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  We used pie charts and bar graphs to see the data distribution in each column, we also used boxplot and bar graphs for comparing the data present in different columns.

- Data Sources and their formats

  The data was collected from amazon.in from the review sections of different products for different electronic devices. These electronic devices consisted of laptops, phones, headphones, smart watches, professional cameras, printers, monitors, home theater, router. The data was collected with the help of browser automation with selenium.

- Data Preprocessing Done

  Not much data preprocessing was required as there were only few missing or null values present in the dataset, which were dropped. The entire text reviews were converted into a matrix of token counts with the help of CountVectorizer which was imported from the scikit learn library. TF-IDF, that is, term frequency inverse document frequency was also used with the help of TfidfTransformer which was also imported from scikit learn library. From nltk corpus stopwords were imported and used to filter out stopwords from the data.

- Hardware and Software Requirements and Tools Used

  All of the work in this project was done on Jupyter notebook. We used pandas and NumPy for working on data and using all the basic mathematical functions on it. We also used matplotlib.pyplot and seaborn libraries for data visualization. Also imported countvectorizer, tf-idf from sklearn library and stopwords from nltk corpus.

Model/s Development and Evaluation

- Testing of Identified Approaches (Algorithms)

  Listing down all the algorithms used for the training and testing.

  We used Multinomial NB, Decision Tree Classifier and Random Forest Classifier models for building our model.

- Run and Evaluate selected models

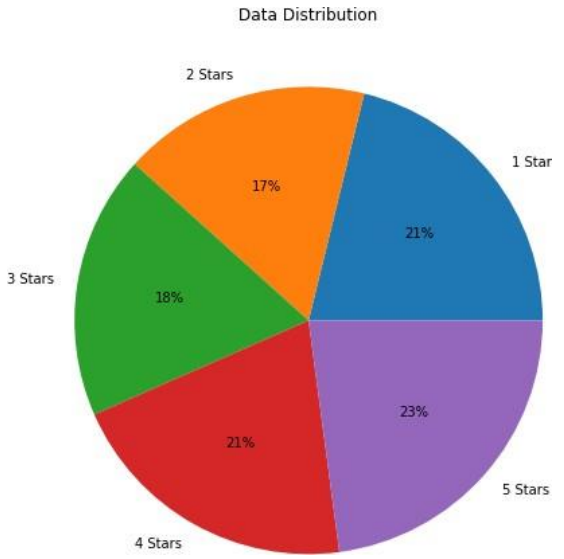
  We tried different NLP processes for data cleaning like countvectorizer, tf-idf, etc and then trained the data in our selected models. These selected models were Multinomial NB, Decision Tree Classifier and Random Forest Classifier.

- Key Metrics for success in solving the problem under consideration

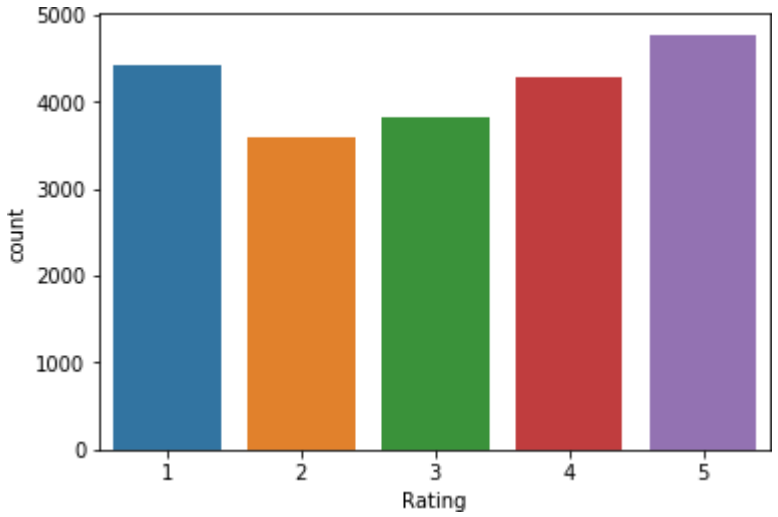
  For evaluation of our models, we used accuracy score, f1 score, precision and recall. To present this we used classification report

- Visualizations

  We used pie charts and bar graphs to see the data distribution in each column, we also used boxplot and bar graphs for comparing the data present in different columns.



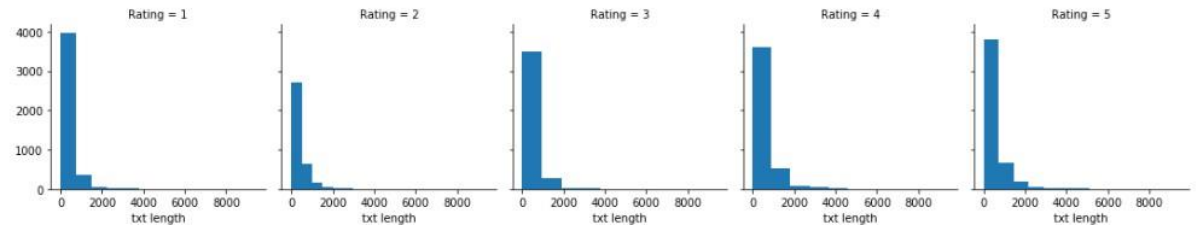

```
vals = [df.Rating[df.Rating==1].count() , df.Rating[df.Rating==2].count(), df.Rating[df.Rating==3].count(),
        df.Rating[df.Rating==4].count(), df.Rating[df.Rating==5].count()]
plt.figure(figsize = (15, 8))
label = ['1 Star' , '2 Stars', '3 Stars', '4 Stars', '5 Stars']
plt.pie(vals , labels=label , autopct = '%1.0f%%')
plt.title('Data Distribution')
```

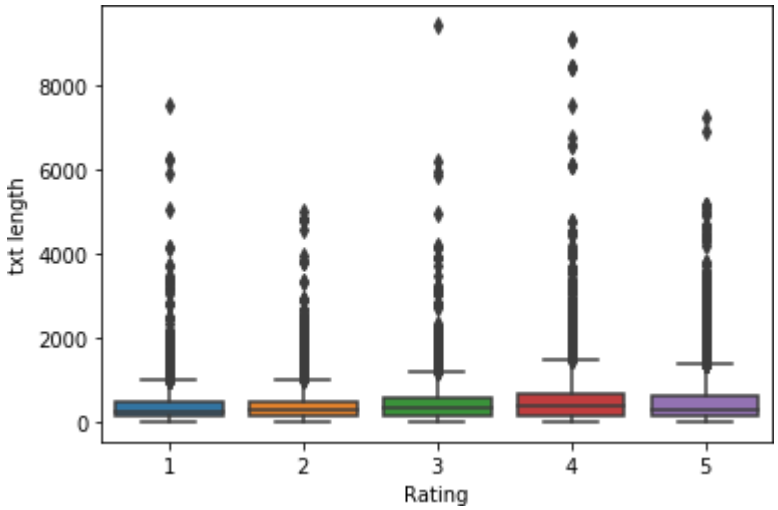



```
sns.countplot(df['Rating'])
```

```
g = sns.FacetGrid(df, col='Rating')
g.map(plt.hist,'txt length')
```

```
<seaborn.axisgrid.FacetGrid at 0x1af8d045088>
```






```
sns.boxplot(data=df, x='Rating', y='txt length')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1af8e94dd88>
```

- Interpretation of the Results

  While collecting and processing models we found that most of the users gave 5,4 or 1 star ratings, while comparably few customers gave 2 or 3 stars ratings. This was evident from the f1 scores of different models for different ratings. The f1 score was usually higher in 1 or 5 star ratings compared to other star ratings.

CONCLUSION

- ## Key Findings and Conclusions of the Study

  In this project we learned about the review and rating systems of online stores.

  While collecting the data from the websites with the help of an automated browser using selenium, we learned how most of the reviews are either 5, 4 stars, i.e., Very Good or Good or they were 1 stars, i.e., Bad. We did find 2 stars and 3 stars reviews but they were not as popular as 5,4 or 1.

  We created a model which could help in predicting the star ratings for the product after processing the review for the said product. We learned about various NLP techniques like count vectoriser, tf-idf, etc.

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

  While working on this project I learned about review and rating systems of online stores and how with the help of machine learning and NLP we can help in improving the performance and services of e-stores, which can help them in increasing their business and customer satisfaction.