Rating Prediction Project

Submitted by:

Rahul singh

ACKNOWLEDGMENT

Thanks for giving me the opportunity to work in FlipRobo Technologies as Intern and would like to express my gratitude to Data Trained Institute as well for training me in Data Science Domain. This helps me to do my projects well and understand the concepts.

Dataset – FlipRobo Tech

Resources used – Google, GitHub, Blogs for conceptual referring.

INTRODUCTION

This is a Machine Learning Project performed on customer reviews. Reviews are processed using common NLP techniques.

Millions of people use Amazon and Flipkart to buy products. For every product, people can rate and write a review. If a product is good, it gets a positive review and gets a higher star rating, similarly, if a product is bad, it gets a negative review and lower star rating. My aim in this project is to predict star rating automatically based on the product review.

The range of star rating is 1 to 5. That means if the product review is negative, then it will get low star rating (possibly 1 or 2), if the product is average then it will get medium star rating (possibly 3), and if the product is good, then it will get higher star rating (possibly 4 or 5).

This task is similar to Sentiment Analysis, but instead of predicting the positive and negative sentiment (sometimes neutral also), here we need to predict the rating.

PROBLEM STATEMENT

The rise in e-commerce has brought a significant rise in the importance of customer reviews. There are

hundreds of review sites online and massive amounts of reviews for every product. Customers have changed their way of shopping and according to a recent survey, 70 percent of customers say that they use rating filters to filter out low rated items in their searches.

The ability to successfully decide whether a review will be helpful to other customers and thus give the product more exposure is vital to companies that support these reviews, companies like Google, Amazon, Flipkart etc.

There are two main methods to approach this problem. The first one is based on review text content analysis and uses the principles of natural language process (the NLP method). This method lacks the insights that can be drawn from the relationship between costumers and items. The second one is based on recommender systems, specifically on collaborative filtering and focuses on the reviewer's point of view.

DATA COLLECTION PHASE

We have to scrape at least 20000 rows of data. We can scrape more data as well, it's up to us. More the data better the model. In this section you need to scrape the reviews of different laptops, Phones, Headphones, smart watches, Professional Cameras, Printers, monitors, home theatre, router from different e-commerce websites.

Basically, we need these columns:

1) reviews of the product.

2) rating of the product.

Fetch an equal number of reviews for each rating, for example if we are fetching 10000 reviews then all ratings 1,2,3,4,5 should be 2000. It will balance our data set. We have to convert all the ratings to their round number as there are only 5 options for rating i.e., 1,2,3,4,5. If a rating is 4.5 convert it 5.

Analytical Problem Framing

- Data Sources and their formats
- After collecting the data, you need to build a machine learning model. Before model building do all data pre-processing steps involving NLP. Try different models with different hyper parameters

and select the best model. Follow the complete life cycle of data science. Include all the steps mentioned below:

1. Data Cleaning
2. Exploratory Data Analysis and Visualization
3. Data Pre-processing
4. Model Building
5. Model Evaluation
6. Selecting the Best classification model

- ▶ Importing the necessary libraries/dependencies
- ▶ Checking dataset dimensions and null value details
- ▶ Taking a look at various label categories using the Unique method
- ▶ Performing data cleaning and then visualization steps
- ▶ Making Word Clouds for loud words in each label class
- ▶ Handling the class imbalance issue manually and fixing it
- ▶ Converting text into vectors using the TF-IDF Vectorizer
- ▶ Splitting the dataset into train and test to build classification models
- ▶ Evaluating the classification models with necessary metrics

Model/s Development and Evaluation

The complete list of algorithms that were used in training and testing the classification model are listed below:

1. Logistic Regression
2. Linear Support Vector Classifier
3. Random Forest Classifier
4. Bernoulli Naïve Bayes
5. Multinomial Naïve Bayes
6. Stochastic Gradient Descent Classifier
7. LGBM Classifier

CONCLUSION

Key findings of the study: In this project I have collected data of reviews and ratings for different products from amazon.in and flipkart.com. Then I have done different text processing for reviews column and chose equal number of text from each rating class to eliminate problem of imbalance. By doing different EDA steps I have analyzed the text. We have checked frequently occurring words in our data as well as rarely occurring words. After all these steps I have built function to train and test different algorithms and using various evaluation

metrics I have selected Random Forest Classifier as our final model. Finally by doing hyperparameter tuning we got optimum parameters for our final model. And finally we got improved accuracy score for our final model.

Limitations of this work and scope for the future work: As we know the content of text in reviews is totally depends on the reviewer and they may rate differently which is totally depends on that particular person. So it is difficult to predict ratings based on the reviews with higher accuracies. Still we can improve our accuracy by fetching more data and by doing extensive hyperparameter tuning.

Areas of improvement:

I. Less time complexity

II. More computational power can be given

III. More accurate reviews can be given

IV. Many more permutations and combinations in hyper parameter tuning can be used to obtain better parameter list

Final Remarks: After applying the hyper parameter tuning the best accuracy score obtained was 72.33278955954323% which can be further improved by obtaining more data and working up through other parameter combinations.

We were able to create a rating prediction model that can be used to identify rating details just by evaluating the comments posted by a customer.