Product Review Rating Prediction Using NLP

Submitted by :
Prathamesh Nayak

INTRODUCTION

- Research has shown that consumer online product ratings reflect both the customers' experience with the product and the influence of others' ratings.
- The opinion information is very useful for users and customers. Businesses can also use the opinion information to design better strategies for production and marketing.
- A recent survey (Hinckley, 2015) revealed that 67.7% of consumers are effectively influenced by online reviews when making their purchase decisions.
- More precisely, 54.7% recognized that these reviews were either fairly, very or absolutely important in their purchase decision making.
- Searching and comparing text reviews can be frustrating for users as they feel submerged with information.

INTRODUCTION

 The star-rating, i.e., stars from 1 to 5 on online platform, rather than its text content gives a quick overview of the product quality.

 The overall star ratings of the product reviews may not capture the exact polarity of the sentiments.

 <u>For instance, a user may rate a product as good and assign a 5-star score while another user may write the same comment and give only 3 stars.</u>

 The question that arises is how to successfully predict a user's numerical rating from its review text content. One solution is to rely on <u>supervised machine learning techniques such as text classification</u> which allows to automatically classify a document into a fixed set of classes after being trained over past annotated data.

PROBLEM STATEMENT

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 rating.

So we, we have to build an application which can predict the rating by seeing the review.

Web Scraping Details

- Web Scraping done using selenium webdriver.

- Data for different product like smartphones, laptops, routers is scraped.

- Data scraped from amazon.in & Flipkart.com

- Around 95000 product reviews are scrap for this project.

Exploration of Target Variable Ratings
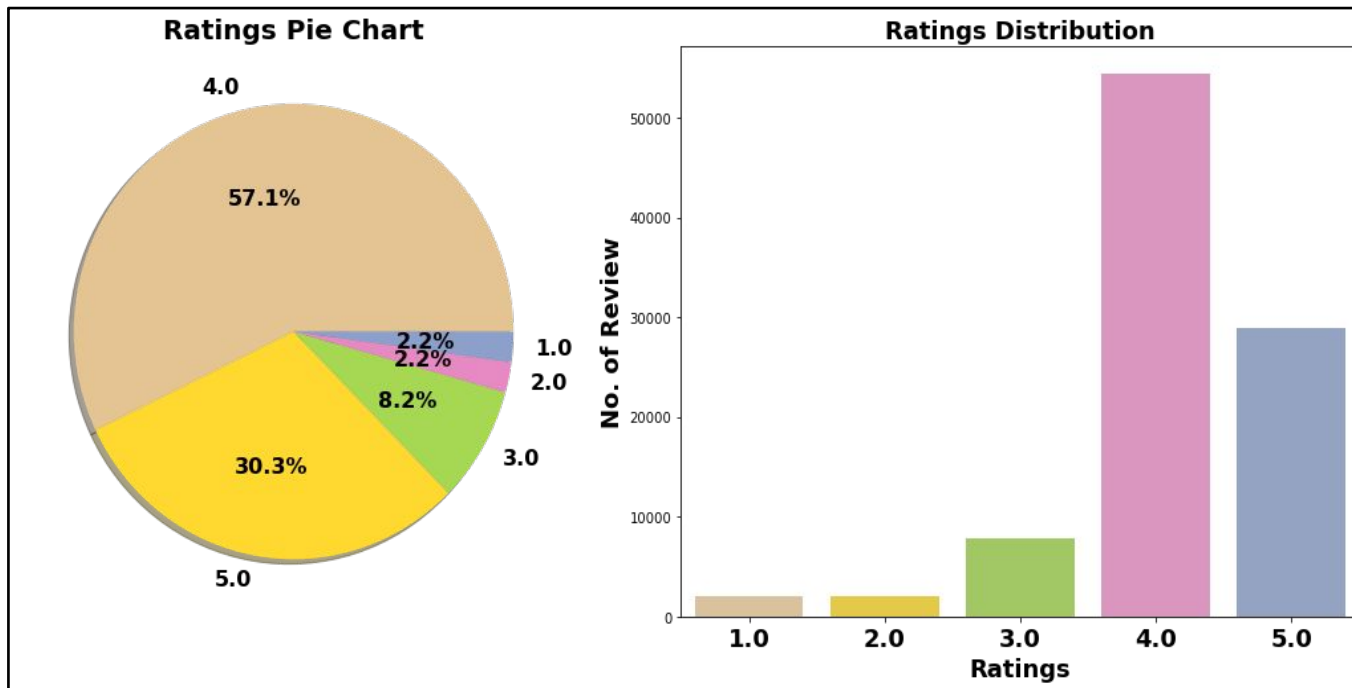
Comment:

1. Around 57.1% customer given 4- star rating followed by 2.2% customer given lowest 1-star rating.

2. Average Rating is 4.11.

Data Pre Processing

- Convert the text to lowercase

- Remove the punctuations, digits and special characters

- Tokenize the text, filter out the adjectives used in the review and create a new column in dataframe

- Remove the stop words

- Stemming and Lemmatization

- Applying Text Vectorization to convert text into numeric

Word Cloud for getting word sense

- Word Cloud is a visualization technique for text data wherein each word is picturized with its importance in the context or its frequency.

- The more commonly the term appears within the text being analysed, the larger the word appears in the image generated.

- The enlarged texts are the greatest number of words used there and small texts are the smaller number of words used.

Word Cloud for Rating 1

Word Cloud for Rating 2

Word Cloud for Rating 3

Word Cloud for Rating 4

Word Cloud for Rating 5

Web Scraping Library used

```python
#Importing required libraries
import pandas as pd
import selenium
from selenium import webdriver
from selenium.common.exceptions import NoSuchElementException , StaleElementReferenceException
import time
import warnings
warnings.filterwarnings('ignore')
```

Text Mining Library used

```python
#Importing required libraries
import re
import string
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import SnowballStemmer, WordNetLemmatizer
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from wordcloud import WordCloud
```

Machine Learning model building Library used

```python
#Importing Machine learning Model library
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import MultinomialNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.model_selection import train_test_split,cross_val_score
from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
```

Machine Learning Model Building

The different classification algorithm used in this project to build ML model are as below:

❖ Random Forest classifier

❖ Decision Tree classifier

❖ Logistics Regression

❖ AdaBoost Classifier

❖ Gradient Boosting Classifier

Machine Learning Evaluation Matrix

- Random Forest Classifier gives maximum accuracy score.

- Hyper parameter Tuning is perform over this best model.

```
GCV.best_params_

{'criterion': 'gini', 'max_features': 'log2', 'n_estimators': 75}
```

Final ML Model

```
Final Random Forest Classifier Model
Accuracy Score :
 0.7397835907133102


Confusion matrix of Random Forest Classifier :
[[    0     3     8   191   198]
 [    0    84    17   216    59]
 [    0    15   208  1026   266]
 [    0    74   103  9650  1166]
 [    0    39    89  1484  4142]]


Classification Report of Random Forest Classifier
              precision    recall  f1-score   support

         1.0       0.00      0.00      0.00       400
         2.0       0.39      0.22      0.28       376
         3.0       0.49      0.14      0.21      1515
         4.0       0.77      0.88      0.82     10993
         5.0       0.71      0.72      0.72      5754

    accuracy                           0.74     19038
   macro avg       0.47      0.39      0.41     19038
weighted avg       0.70      0.74      0.71     19038
```

5-fold Cross validation performed over all model. We can see that Random Forest Classifier gives us good Accuracy and maximum f1 score along with best Cross-validation score. Hyperparameter tuning is applied over Random Forest model and used it as final model.

CONCLUSION

- Successfully developed Machine learning model to predict product review ratings.

- NLTK library used for text Mining.

- Random forest classifier model is best model with accuracy score of 73.95%.