Project presentation on :-

RATINGS PREDICTION PROJECT

SUBMITTED BY :

Alfa Wesley Sirra

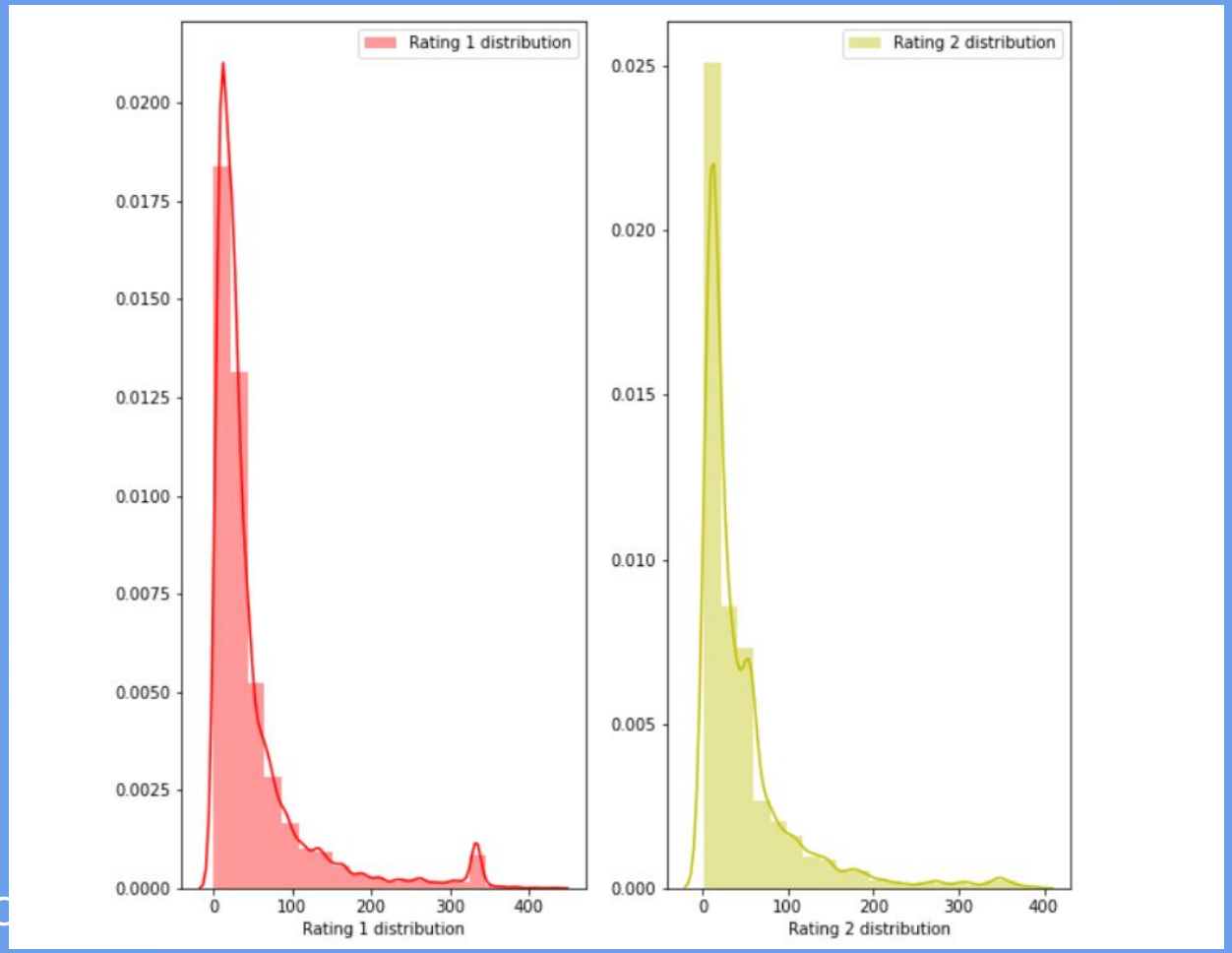
Table Of Contents :-

INTRODUCTION

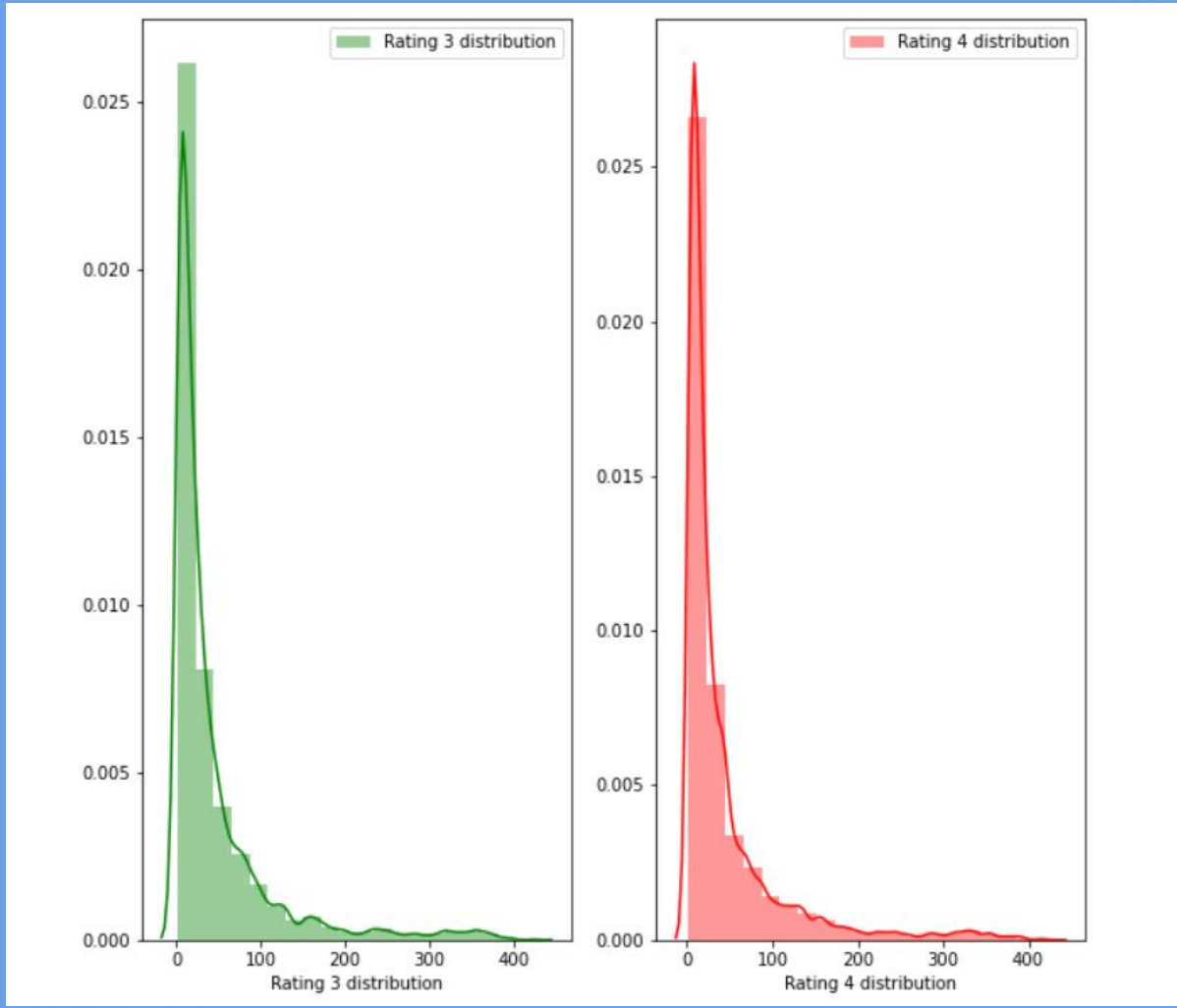
Problem statement and understanding

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.

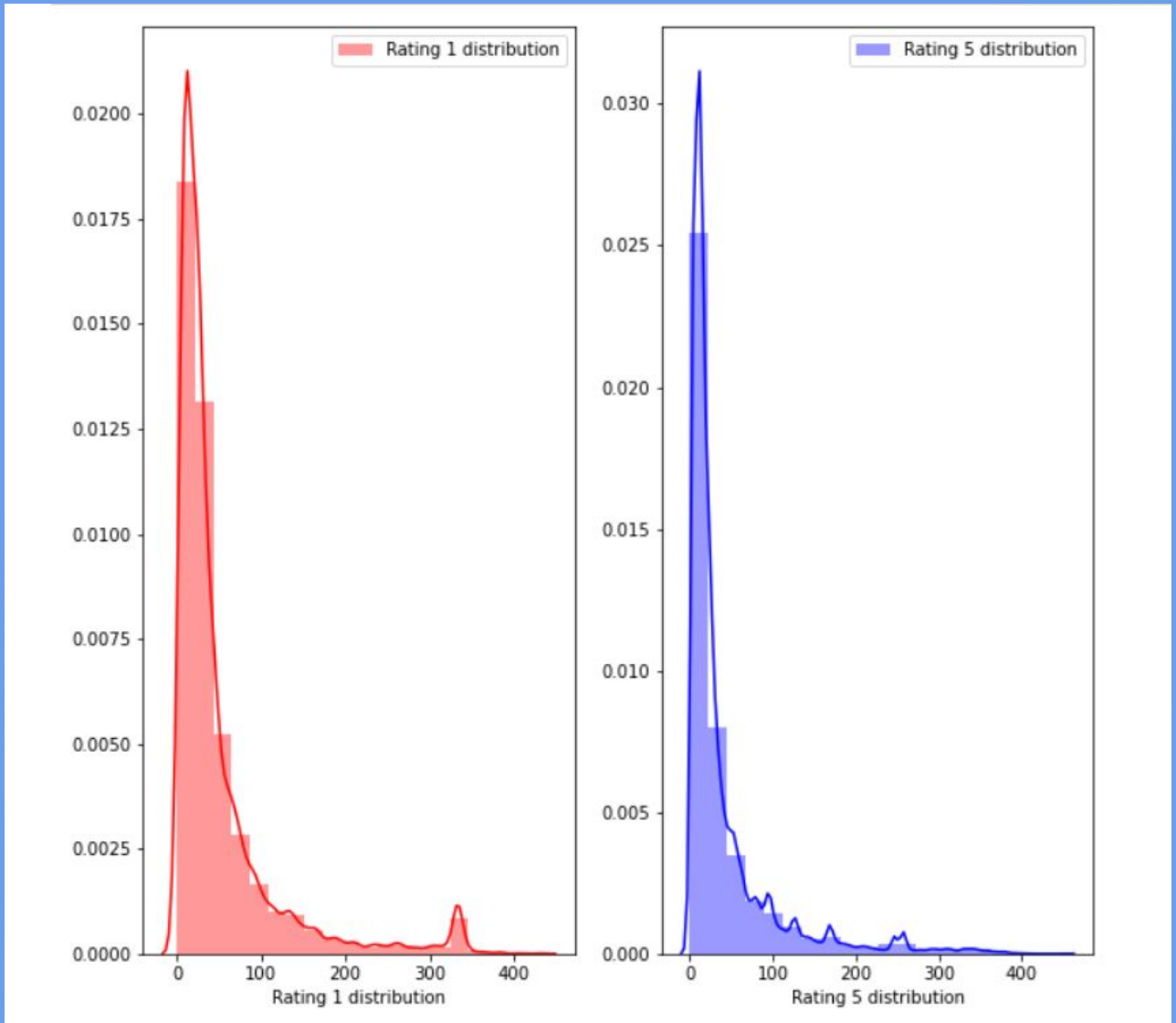
EDA steps and Visualization



Rating 1 and and Ra



Rating 3 and and Rating 4 distribution after cleaning the reviews:



Rating 1 and Rating 5 distribution after cleaning reviews:

Getting sense of review Loud words in Rating 1:

Getting sense of review Loud words in Rating 1:

Getting sense of review Loud words in Rating 3:

getting sense of review Loud words in Rating 4:

getting sense of review Loud words in Rating 5:

Data Preprocessing Done

We first looked for the null values present in the dataset. We noticed that there were no null values present in our dataset. Then we performed text processing. Data usually comes from a variety of sources and often in different formats. For this reason transforming your raw data is essential. However, this is not a simple process, as text data often contains redundant and repetitive words. This means that processing the text data is the first step in our solution. The fundamental steps involved in text preprocessing are, Cleaning the raw data Tokenizing the cleaned data.

Preprocessing involved the following steps:

- Removing Punctuations and other special characters
- Removing Stop Words
- Stemming and Lemmatising
- Applying tfidf Vectorizer
- Balancing the dataset through smote technique

- Some very large length comments can be seen, in our dataset. These pose serious problems like adding excessively more words to the training dataset, causing training time to increase and accuracy to decrease! Hence, a threshold of 400 characters will be created and only comments which have length smaller than 400 will be used further.

- Hence, after removing comments longer than 400 characters, we are still left with 115893 comments, which seems enough for training purposes.

Set of assumptions related to the problem under consideration

- By looking into the target variable label we assumed that it was a Multiclass classification type of problem.

- We observed that dataset was imbalance so we will have to balance the dataset for better outcome.

Model Dashboard

	Model	Accuracy_score	Cross_val_score
0	KNeighborsClassifier	42.772643	57.831708
1	DecisionTreeClassifier	56.759088	57.940008
2	XGBClassifier	58.311768	63.480543
3	RandomForestClassifier	60.973506	63.870033
4	AdaBoostClassifier	51.608133	62.243298
5	MultinomialNB	54.799754	62.524411
6	GradientBoostingClassifier	56.487985	63.231706
7	BaggingClassifier	57.966728	61.402893
8	ExtraTreesClassifier	60.862600	63.751793

We observe that Random forest classifier is giving is best results so we save it as our final model.

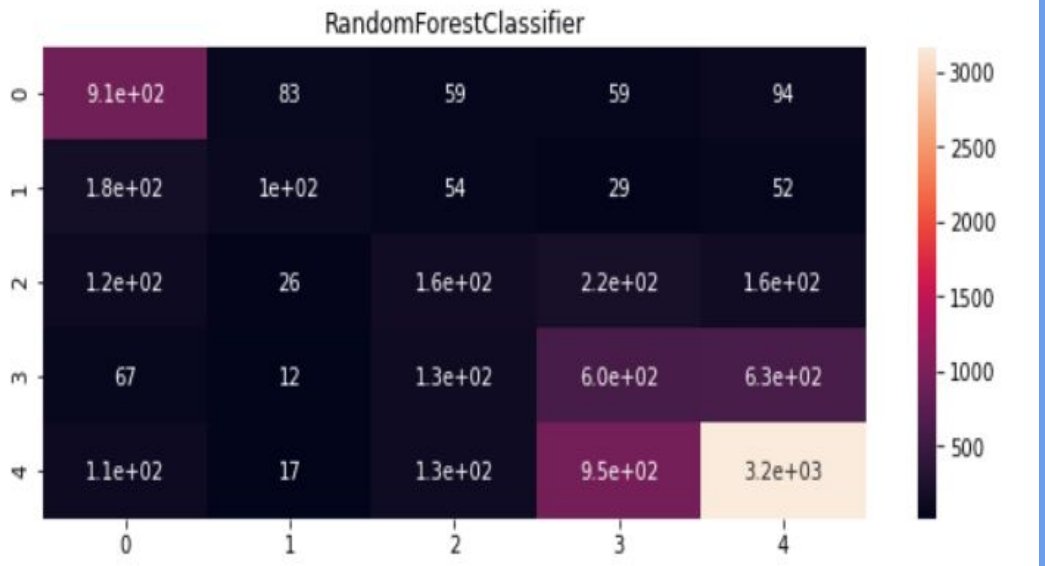
Finalized Model

Accuracy_score = 0.6097350585335798

Cross_Val_Score = 0.6387003315130022

classification_report

	precision	recall	f1-score	support
1	0.66	0.76	0.71	1209
2	0.42	0.25	0.31	412
3	0.30	0.23	0.26	678
4	0.32	0.42	0.37	1438
5	0.77	0.72	0.75	4378
accuracy			0.61	8115
macro avg	0.50	0.48	0.48	8115
weighted avg	0.62	0.61	0.61	8115



We interpreted that Random forest classifier model was giving us the best results with the accuracy score of 60.97 and comparitively better f1-score so we saved it as our final model.

Conclusion

In this project we have tried to detect the Ratings in commercial websites on a scale of 1 to 5 on the basis of the reviews given by the users. We made use of natural language processing and machine learning algorithms in order to do so. We interpreted that Random forest classifier model is giving us best results.

Acknowledgement

I would like to express my special thanks of gratitude to the sources Medium, Towards Data Science, Stack Overflow, which helped me to accomplish this project.