

Ratings Prediction

Submitted by:

Nikhil Singh Rana

**ACKNOWLEDGMENT**

The satiation that accompanies the successful completion of the project would be incomplete without the mention of the people who made it possible

I would like to take the opportunity to thank and express my deep sense of gratitude to my data trained academy mentors for providing their valuable guidance at all stages of the study of my data scientist course, their advice and constructive suggestion through which I have gained this much skills that I can complete this project.

I have taken the help of my previous projects which I had done in my training phase with data trained academy and also refered google for some line of codes .

**INTRODUCTION**

* Business Problem Framing

In the modern day, public discussion and critiquing of products and services occurs beyond dedicated mediums, and now also takes place in the realm of social media, too.

Online Product reviews are currently found on Amazon, Flipkart, and Myntra etc. The majority of reviewers gave a score between 3 and 5, so if a new customer browses online reviews on any of the previously mentioned review sites, they may consider a product from amazon or flipkart.

What if they already made up their mind from hearing what a friend had to say? Potential customers, could have their product choice be influenced by a tweet. Opinions are shared constantly on social media platforms, and are read by their followers. The knowledge, of what these followers think about the product, from reading these online posts, could help us better understand the general public's perception of the product.By using sentiment analysis, on existing product reviews from amazon and flipkart, I created a model that can quantify on a scale of 1-5, how the author of a reviews on amazon or flipkart, feels about the product, and as a result, also how the readers think about the particular product. If a review classifies to be less than a score of 3, this review could be looked into, find out why they had a negative opinion of the product, and in return fix the problem.

* Conceptual Background of the Domain Problem

A very simple definition that is commonly found online for Natural Language Processing (NLP) is that it is the application of computation techniques on language used in the natural form, written text or speech, to analyse and derive certain insights from it. These insights could vary quite widely, from understanding the sentiment in a piece of text to identifying the primary subject in a sentence.

The complication of the analysis can also range from simple to very complex. For instance you may want to categorise pieces of text, say from Twitter data, as having positive sentiment or negative sentiment. On the other hand you may want to extract product suggestions/complaints from product review data from a large dataset to figure out new product launch strategies. The work being done on chatbots, to understand text queries and respond to them; spam detection, etc. falls in the ambit of NLP.

This field can encompass many applications and thus many techniques, which makes it a fairly large topic. For this particular tutorial we will be discussing in detail a business case application of NLP, to derive practical and actionable insights in the e-commerce domain.

* Review of Literature

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 and flipkart.

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. Use of the user’s similarity matrix and applying neighbors analysis are all part of this method. This method ignores any information from the review text content analysis.

* Motivation for the Problem Undertaken

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

1. Web Scraping:

I have scraped almost 5000 reviews of 1 start, 2star, 3star, 4star, 5star ratings from the Amazon.com and Flipkart.com by using selenium and beautiful soup libraries.

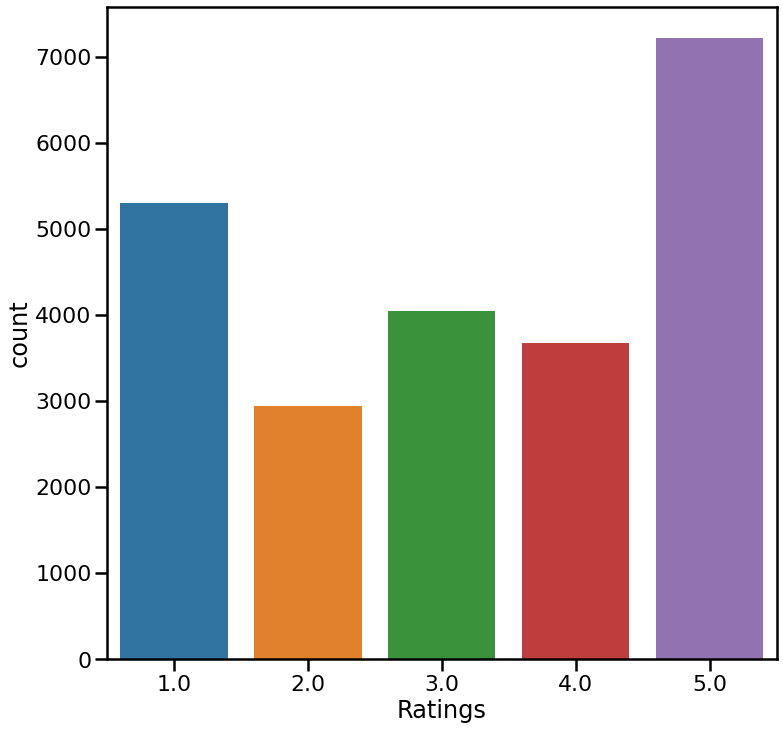
After scraping I saved it in a excel file where there is 2 column one is review and other is ratings.

1. Early EDA and Cleaning:

The initial shape of the dataset was (23175,2).

This project entailed the use of classification models, and for reliable results, I had to remove reviews to undo class imbalance. Using this visualisation I saw that were much less reviews with a score of 2 compared to reviews with a score of 1,3, 4, and 5. To combat this imbalance, I randomly removed reviews with scores of 1, 3, 4, and 5, to match with 1 (2936 reviews).

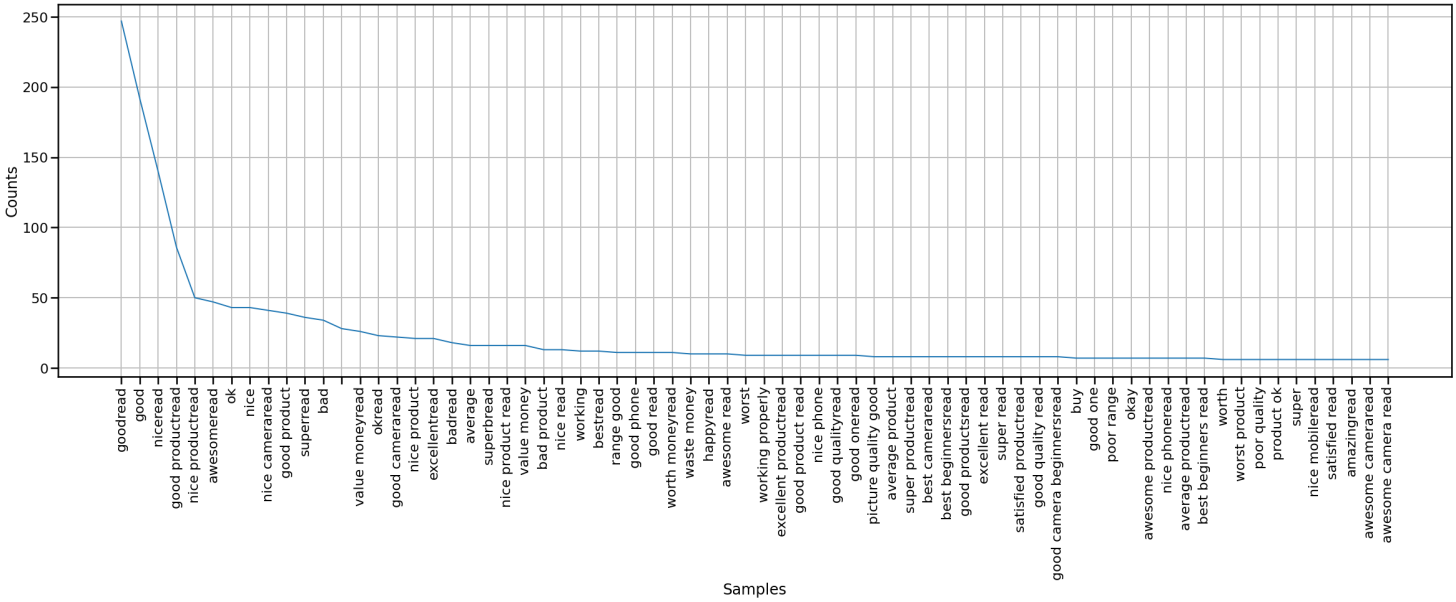


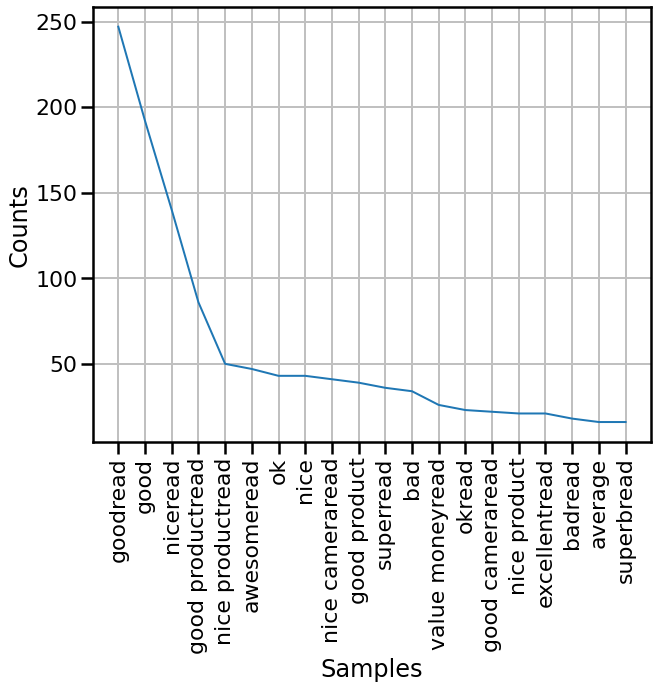


1. Further EDA:

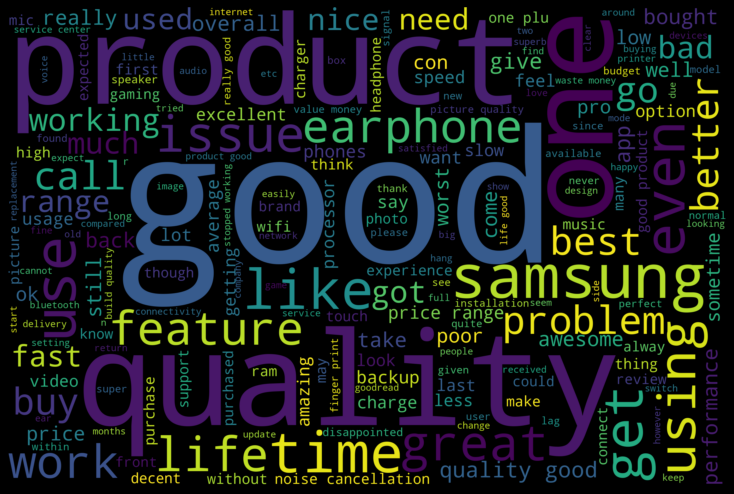
The cleaned dataset had a shape of (14689,2). I started with some analysis on the text columns; review and review summary.

Using the FreqDist function in the ntlk library I plotted a graph with the most frequent words and phrases in both columns. Stopwords were removed to capture the more meaningful words.





I had noticed a lot of the most frequent words in the review text happened to be words with no sentimental impact, so I iteratively removed unmeaningful words such as 'phone', ‘use’, battery etc. I did this as a precaution, as some of these words may impact my model accuracies.



* Data Sources and their formats

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.

I have scrape the reviews of different laptops, Phones, Headphones, smart watches, Professional Cameras, Printers, monitors, Home theater, router from different e-commerce websites.

* Data Preprocessing Done

Preprocessing is a key step in any analysis and in this project as well.

1. Stop Words:

Firstly, I removed all the stop words and different symbols in the review and unmeaningful words.

1. Stemmation and lemmitisation:

To narrow down the feature words I applied stemmation and lemmitisation to both the reviews and review summaries.

1. TfID:

Stemmation had broken down some words into words that don't exist, whereas lemmitisation had simplified adjectives and verbs to their root form. I chose to continue with the lemmitised version of the texts for further processing. Prior to vectorising the current dataset, I did a train, test split to save the test data for after modelling. Using the lemmed texts for review I used TF-IDF vectorisation with an ngram range of 2, leaving me with a vectorised dataset with 33 words and phrases . I then saved the x and y train data in separate csv files for modelling.

* Hardware and Software Requirements and Tools Used

Technology used:

* **Python**
* **Pandas**
* **Numpy**
* **Matplotlib**
* **Seaborn**
* **NLTK**
* **Scrapy**
* **Scikit-Learn**
* **Keras**
* **Tensorflow**
* **Streamlit**
* **Heroku**

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

The first step of the ML process is to look at the data to see what’s there and to begin thinking about the other steps of the ML process, such as model type and featurization.

Often practitioners of (non-real-world) machine learning dive into a problem without thinking hard about the practical use of their ML model. This is a mistake, because the choice of use case can help determine how you structure the problem and solution, including the following:

How to encode the target variable (for example, binary versus multiclass versus real value)

Which evaluation criterion to optimize

What kinds of learning algorithms to consider

Which data inputs you should and should not use

So before you get started with ML modeling, you first need to determine what real-world use case you want to solve with this dataset.

For each of three possible use cases, you’ll consider the following:

* Why would the use case be valuable?
* What kind of training data would you need?
* What would an appropriate ML modeling strategy be?
* What evaluation metric should you use for your predictions?
* Is the data you have sufficient to solve this use case?
* Testing of Identified Approaches (Algorithms)

I have created .py files; Classifiction.py and Ensemble.py with classes, that contain functions to simplify the modelling process, and to neaten up the modelling notebook.

For the majority of models I created, I applied hyperparameter tuning, where I started with a broad range of hyperparameters, and tuned for optimal train accuracy and validation accuracy.

I focused on 3 factors of defining a good model:

Good Validation Accuracy

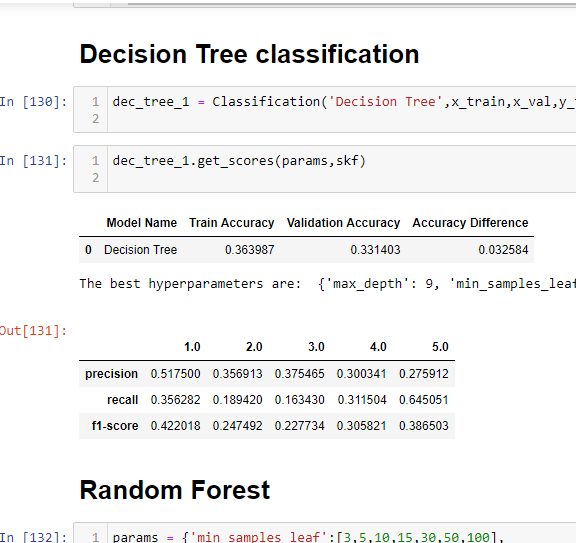
Good Training Accuracy

Small Difference between Training and Validation Accuracy

* Run and Evaluate selected models

I have used Following Algorithms and model:

* Decision Tree
* Random forest  
  SvM
* KNN



* Key Metrics for success in solving problem under consideration

Looking at the precision, recall, and f1 score, I also noticed the scores were higher around scores of 1 and 5, lower for 2, 3, and 4. This shows that the models performs well on more extreme opinions on reviews than mixed opinions.

**CONCLUSION**

* Key Findings and Conclusions of the Study

Key findings are that, the EDA process helps us to understand the problem well through visualizations and data cleaning, this will further help us in model building purpose and gaining the maximum accuracy score.

Through EDA process we can see that there are some columns which are not needed and there are some outliers to be removed so that the accuracy should be maintained.

Also there are high correlation between some columns . Therefore we can visualize and check the relationship between that columns.

* Learning Outcomes of the Study in respect of Data Science

Firstly visualization helps us in various forms as follows:

* **Communicate Findings in Constructive Ways**

## Understand Connections Between Operations and Results

## Interacting With Data

## Create New Discussion

## Secondly, after applying algorithms we can see which algorithm performs better and gives better prediction.

## We can also use ensembling techniques to improve the accuracy of the models build.

.

* Limitations of this work and Scope for Future Work
* Model using neural networks - see if better accuracy can be achieved
* Create a working application to test new reviews written by people
* Try a different pre-processing approach and see if model performances change
* Bring in new sources of data to see if there are significant differences on frequent words used

.