Springboard Capstone Project 1

Book Recommender System

Introduction:

Today, the e-commerce industry is growing at an unprecedented rate. Thousands of items are being added to the e-catalog every month and the contents available for users to explore are overwhelming, hence there is a need to shift and prioritize user preferences by analyzing relevant information on items and users. Most of the big e-commerce companies started its online marketplace by selling books but later diversified to sell electronics, apparels, food, toys and jewelry. Success of online book sales has made great impact to the e-commerce industry and new machine learning techniques are being introduced to drive the sales up north.

One of the techniques which can make a meaningful impact to online book sales is the Recommendation system. A Recommendation system helps users to discover items that they may like. The system has become fundamental applications in e-commerce websites by providing suggestions that effectively snip large information so that users are directed toward those items that best meet their needs and preferences. The main goal is to build a system which can make a best suggestion of books

Data Source:

In this project, different recommender system techniques will be utilized to recommend top books for the users by analyzing their ratings. The project will utilize the Book-Crossings dataset which has been publicly shared from University of Freiburg related website. Book-Crossings is a book ratings dataset compiled by Cai-Nicolas Ziegler based on data from bookcrossing.com. It contains 1.1 million ratings of 270,000 books by 90,000 users. The ratings are on a scale from 1 to 10, and implicit ratings are also included. The dataset consists of three tables namely BX-Users, BX-Books and BX-Book-ratings which effectively divides users, books and ratings data in separate tables.

Springboard Capstone Project 1

Book Recommender System

Data Wrangling:

Book Dataset:
As part of Data cleansing, columns such as Image URLs from the book dataset was dropped from the table. Due to some errors in the csv file, there were three records where the values were populated in the incorrect columns. The data manipulation was done to update these incorrect records in such a way that the values were pointing to the right columns. Since the data was compiled in the year 2004, any records having year of publication beyond 2004 was updated to mean year of publication. Additionally, records with null values for Year of publication were populated by mean year of publication.

User dataset:
The User table didn't include Age information for all the Users. Additionally, the Age column had some values which were in the unacceptable range such as beyond 80 and less than 10 years old. In order to capture the right tech-savvy age of the population, Users having age below 10 years were set to 10 years and Users having age beyond 80 was set to 80. The missing ages were replaced by mean age of the Users.

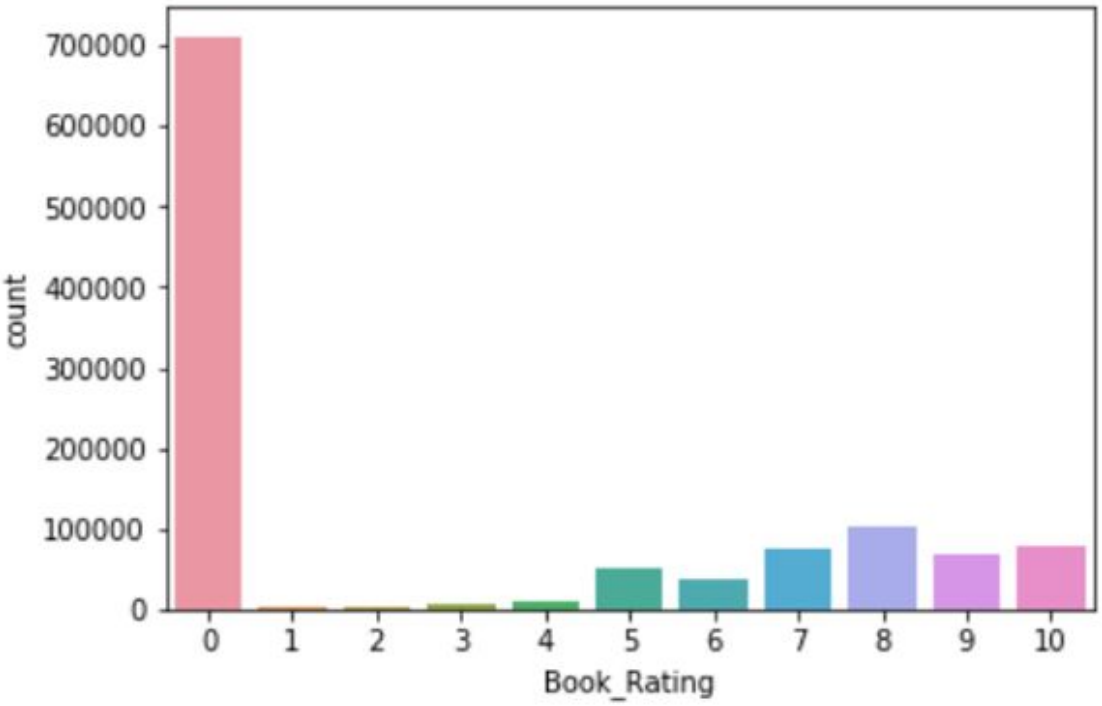
User Ratings table:
The implicit ratings were ignored after visualization as the main intention was to build model based on explicit ratings.

Data Visualization:

In order to capture more insights from the data, a count plot was plotted to capture how the ratings were distributed across all the books.

Springboard Capstone Project 1

Book Recommender System

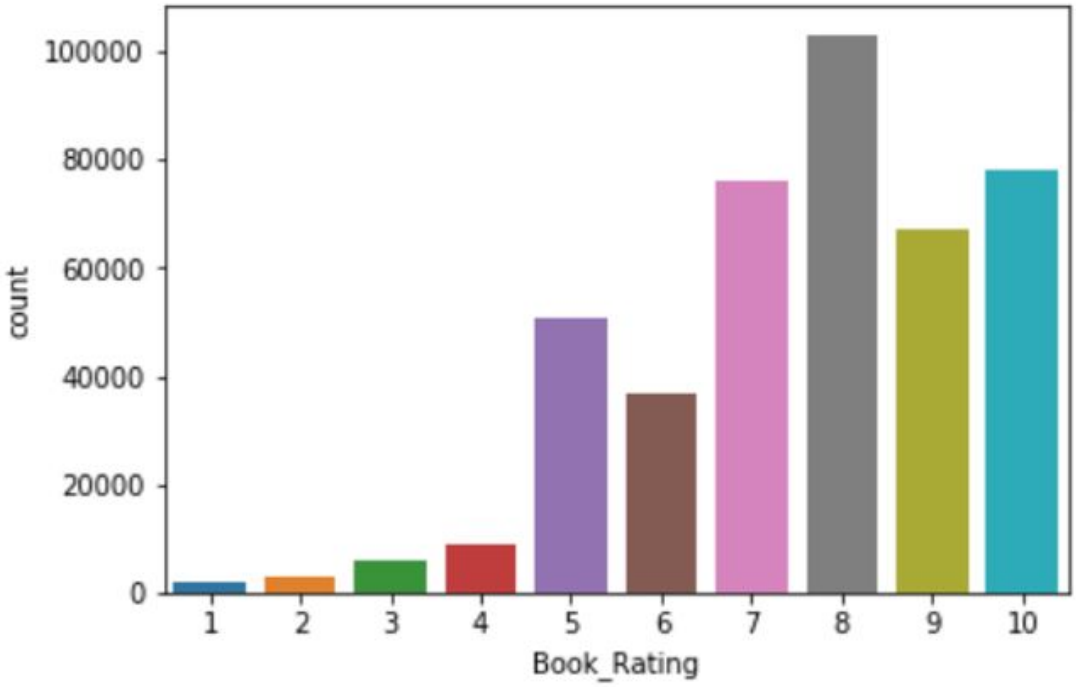


From the countplot, it was clear that the implicit and explicit had unequal representation in the ratings table with books having around 700k implicit ratings recorded out of 1.14M ratings.

Since the focus was on the explicit ratings to model the recommendation system, count plot was plotted to check how it is distributed across.

Springboard Capstone Project 1

Book Recommender System



From the plot, it was clear that the majority of the books had favorable rating which was greater than 7.

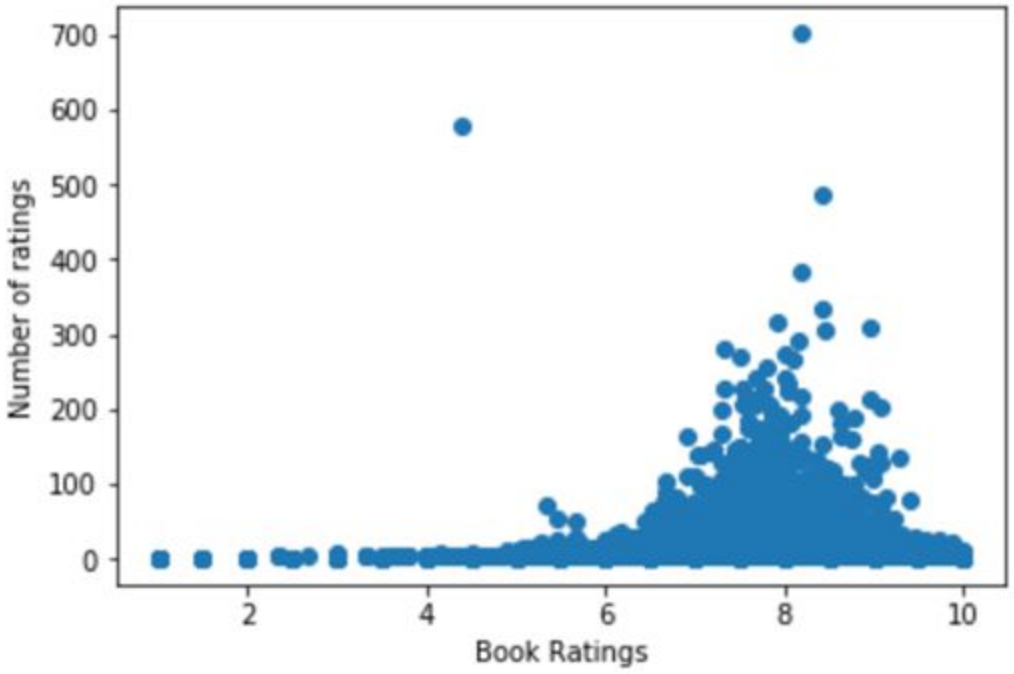
Correlation:

The number of ratings and average ratings for each book was analyzed to check how the ratings correlated.

A scatter plot was plotted to check the distribution of data points.

Springboard Capstone Project 1

Book Recommender System




```
Pearsons correlation: 0.022
```

From the plot, it was clear that there was no correlation between the variables. The Pearsons correlation coefficient backed the assumption by calculating a low coefficient value of 0.022 for the two variables.

Building a collaborative filtering based recommendation system:

There are two types of collaborative filtering namely,

1) Item based CF: This method recommends based on the similarity between items calculated using people's ratings of those items.

2) User based CF: This method identifies users that are similar to the queried user and estimate the desired rating to be the weighted average of the ratings of these similar users.

Springboard Capstone Project 1

Book Recommender System

In both the methods, we had to create a user-item matrix built from the dataset with the users as the rows, the books as the columns, and the rating as the matrix value.

User-tem matrices were created for the training and testing data once the identifiers were converted to sequential integers. As the next step,the cosine similarity of Users and Books was calculated. Cosine similarity measures the cosine of the angle between two vectors projected in a multi-dimensional space. Using this memory based collaborative method, the RMSE value was found to be close to 7.8 which suggested that the method was not ideal for recommendation as it had high variance of the estimator and bias. Overall, Memory-based Collaborative Filtering was easy to implement and produced reasonable prediction quality. However, there are some drawbacks to this approach. Firstly, it can't deal with sparse data. Secondly, it fails to predict for new users/items that doesn't have any ratings in the system.

Building recommendation system using surprise library:

Here the model based collaborative filtering was performed in order to overcome limitations of memory based recommendation system. Model based collaborative filtering uses matrix factorization methods to find the hidden features from the given data. Singular Value Decomposition(SVD) is one such matrix factorization which we use to find the features in the vector space. SVD is an algorithm that decomposes a matrix A into the best lower rank approximation of the original matrix A. It decomposes matrix A into a two unitary matrices and a diagonal matrix.

The SVD model was fitted against the train data, Root Mean Square Error for test data was found to be around 1.6475 which was much better than what we got (7.85) using memory based recommendation system.The hyperparameters was tuned using GridSearchCV method to make prediction more accurate. As a result,

there was slight improvement in RMSE values when the best hyperparameters were chosen.

Top 10 recommendation of books was made to the user id ‘60244’ to check the accuracy of the recommendation. Upon individually looking at the average rating of each recommended book, the recommendation appeared to be well executed.

```
Top 10 book recommendation for User having ID#60244:
The Return of the King (The Lord of the Rings, Part 3)
Divine Secrets of the Ya-Ya Sisterhood: A Novel
The Lion, the Witch, and the Wardrobe (The Chronicles of Narnia, Book 2)
The Hitchhiker's Guide to the Galaxy
Mere Christianity: A revised and enlarged edition, with a new introduction, of the three books, The case for Christianity,
Christian behaviour, and Beyond personality
The BFG
Ender's Game (Ender Wiggins Saga (Paperback))
The Shrinking of Treehorn
Where the Red Fern Grows
Writing Down the Bones
```