**Capstone Project Submission**

**Instructions:**

1. Please fill in all the required information.
2. Avoid grammatical errors.

|  |
| --- |
| **Team Member’s Name, Email, and Contribution:** |
| Contribution Role:     1. **Rajesh Patil** (rajesh.patil775607@gmail.com)  * Data Preparation (Data Cleaning and Feature Engineering) * Exploratory Data Analysis * Got top ten books as per ratings  1. **Chetan Patil ()**  * Outlier detection * Distribution graphs * Which country ha a large number of readers?  1. Mrunal Badgujar ()  * Model Evaluation Metrics * RMSE score with each collaborative filtering approach * Model implementation  1. Sachin Chaudhari ()  * Getting top recommendations for books and ratings * Conclusion * Null value detection |
| **Please paste the GitHub Repo link.** |
| GitHub Link: - https://github.com/Rajesh1patil/Book-recommender-system/blob/main/Book\_recommender\_system.ipynb |
| **Please write a short summary of your Capstone project and its components. Describe the problem statement, your approaches, and your conclusions. (200-400 words)** |

A book recommendation system is a type of recommendation system where we have to recommend similar books to the reader based on his interest. We’ve all recorded data with three different datasets – Book dataset (ISBN, Book-Title, Book-Author, Year-Of-Publication, Publisher, Image URL-S, Image-URL-M, Image-URL-L); Users dataset (User-ID, Location, Rating dataset (User-ID, ISBN, Book-Rating). Providing specific data analysis and predictable n to done with this data. The main objective is to build a predictive recommender model, which could help in predicting – how we can predict the best recommendation for users according to their items approach. This would help us in providing better recommendations items to the right specific user.

As the first step, I performed data preparation (i.e., data cleaning and feature engineering) and the EDA part. The book\_data, users\_data and ratings\_data dataset was imported dataset to get smaller insights from them. And features like book\_title, book author, publisher, year\_of\_publication, age, and book\_rating were important to get some insights. Some of the null values were present in features which data, we replaced with the mean of that particular feature. Deal with misfeatures like book\_title, book\_author, year\_of\_publicatand ion, publisher. Only considering ages between 5-90 we took users’ data to analyze and perform recommendations on it.

After data preparation, build a recommendation system based on popularity (i.e., ratings). These recommendations are usually given to every user irrespective of personal characterization. Merged book\_data dataset and ratings\_explicit. Considering ISBNs that were explicitly rated for this recommendation system.

The third step was to do Collaborative Filtering - Memory based approach was our first trial on train and test dataset which uses the memory of previous users’ interactions to compute users’ similarities based on items they’ve interacted with (i.e. user-based approach) or compute items similarities based on the users that have interacted with them (i.e. item-based approach). Applying cosine similarity to make item-item similarity needs to take the transpose of matrix. This matrix would help in managing train-test matrix. After all views predictions based on similarity, we find recommendation on it based on score. Model-based collaborative filtering algorithms provide item recommendations by first developing a model of user ratings. We use Latent Factor Model called Singular Value Decomposition (SVD). SVD made dimensionality reduction technique in machine learning. SVD is a matrix factorization technique. It made us reduces the number of features of a dataset by reducing the space dimension from N-dimension to K-dimension (where K<N). SVD uses a matrix structure where each row represents a user, and each column represents an item. The elements of this matrix are the ratings that are given to items by users.

Model evaluation metrics are important to distinguish the best collaborative filtering – either by memory-based or model-based approach. The memory-based approach – Cosine Similarity shows RMSE score for item-based CF is 8.00 and for user-based CF it shows 8.00. The score is slightly similar. Model-based collaborative filtering made it better score with the Latent Factor Model called SVD. The score improved to 1.63 for both SVD RMSE and accuracy scores.

The final step was to get a recommendation, Collaborative Filtering- A model-based approach was among best methods to approach the recommendation system for this project. So SVD with RMSE score is the best model with 1.63 for this dataset.

That's how I have accomplished my project work in Book Recommendation System.