Capstone Project
Book Recommendation System

Team

Rahul Kumar Soni, Lakdawala Ali Asgar

Content

- Introduction
- Problem Statement
- Data Summary
- Approach Overview
- Exploratory Data Analysis
- Modelling Overview
- Challenges
- Conclusion

Introduction

Recommender systems are machine learning systems that help users discover new product and services.

A recommendation system helps an organization to create loyal customers and build trust by them desired products and services for which they came on your site.

A book recommendation system is a type of recommendation system where we have to recommend similar books to the reader based on his/her interest. The books recommendation system is used by online websites which provide ebooks like google play books, open library, goodReads, etc.

Problem Statement

To create a book recommendation system for users.

What is recommendation system ?

Recommendation systems are used to gain more user attraction by understanding the user’s taste. These systems have now become popular because of their ability to provide personalized content to users that are of the user’s interest.For eg Netflix suggest the same genre movies to us by understanding our interest/ choice of movies we like similarly Youtube recommends videos to us. There are many different recommendation engines that work backends to make it possible.

Data Summary

Users

- **User-ID - Unique id for each user**
- **Location - in form of town, city and state**
- **Age - Numerical data**

Data Summary

Books

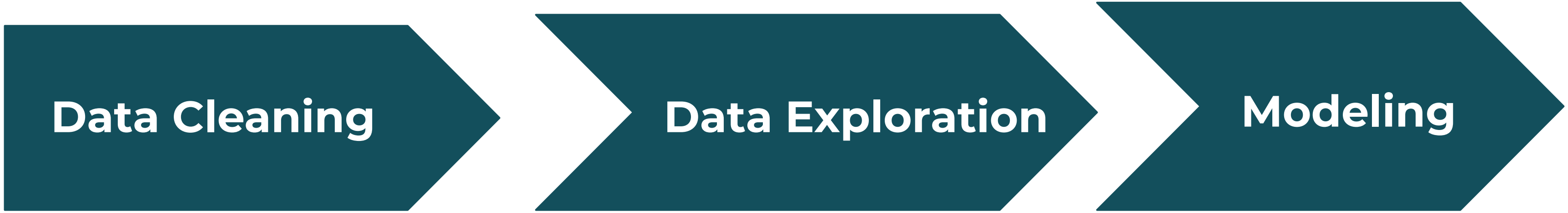
- ISBN : The International Standard Book Number (ISBN) is a unique International Publisher's Identifier number
- Book-Title : Title of the books
- Book-Author : Author of the books
- Year-Of-Publication : Publishing year
- Publisher : A company or person that prepares and issues books for sale
- Image-URL-S,M and L : amazon image url link

Data Summary

Ratings Data

- User-ID Unique id for each user
- ISBN The International Standard Book Number (ISBN) is a unique International Publisher's Identifier number
- Book-Rating : Rating given by user in range [0,10]

Pipeline



Data Cleaning

Data Exploration

Modeling

Understanding and Cleaning

- Null/Missing value analysis and treatment
- Outlier Treatment

Graphical

- Univariate analysis with visualization
- Bivariate Analysis with visualization

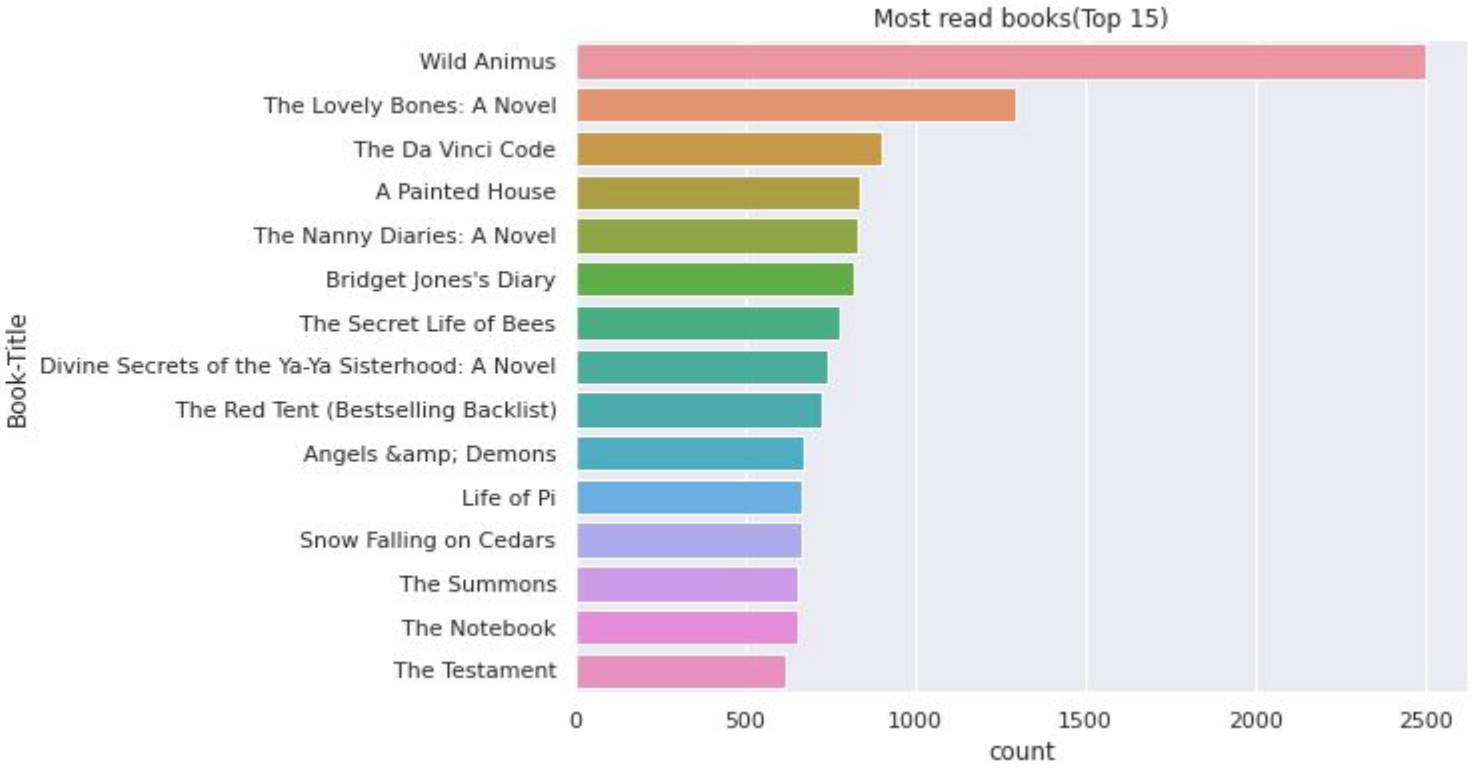
Machine Learning

- **Nearest neighbours**
- **Content based Filtering**
- **Collaborative Filtering model based**

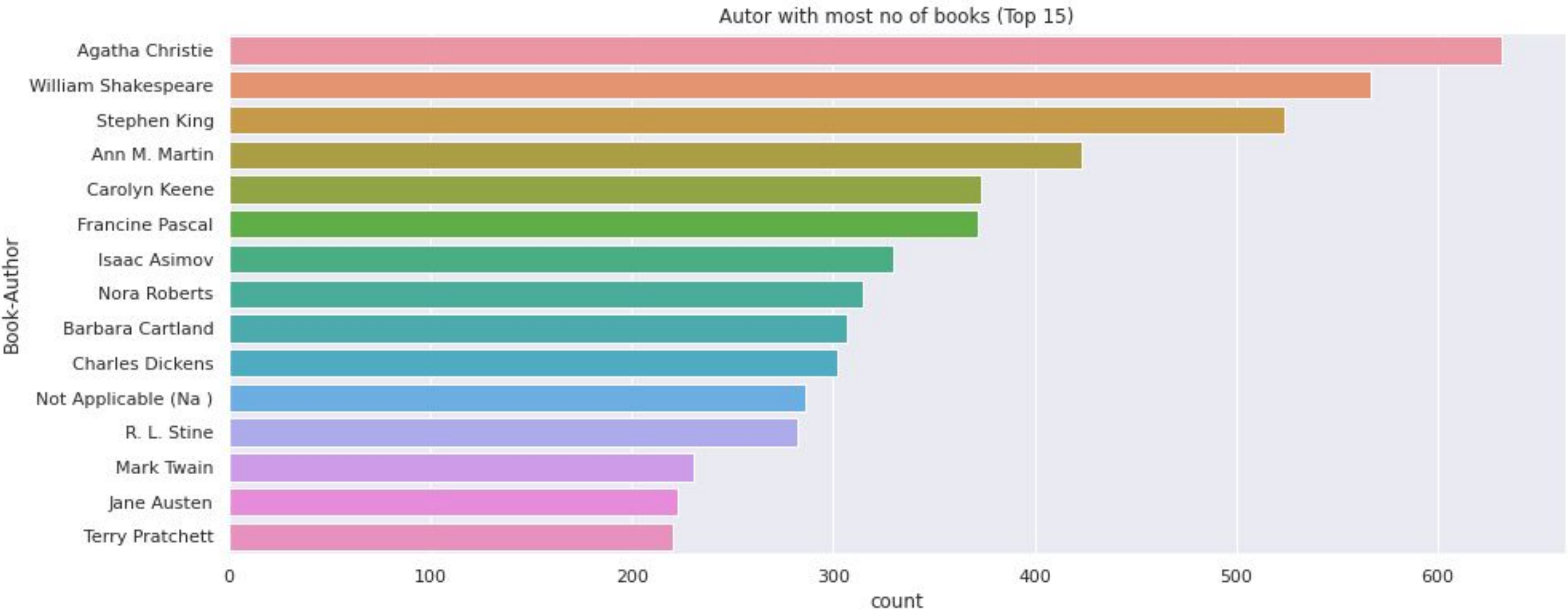
Basic Exploration

- No of book title = 242135
- No of author = 102024
- No of Publisher = 16805
- Most of the books are published in between 1999 to 2002

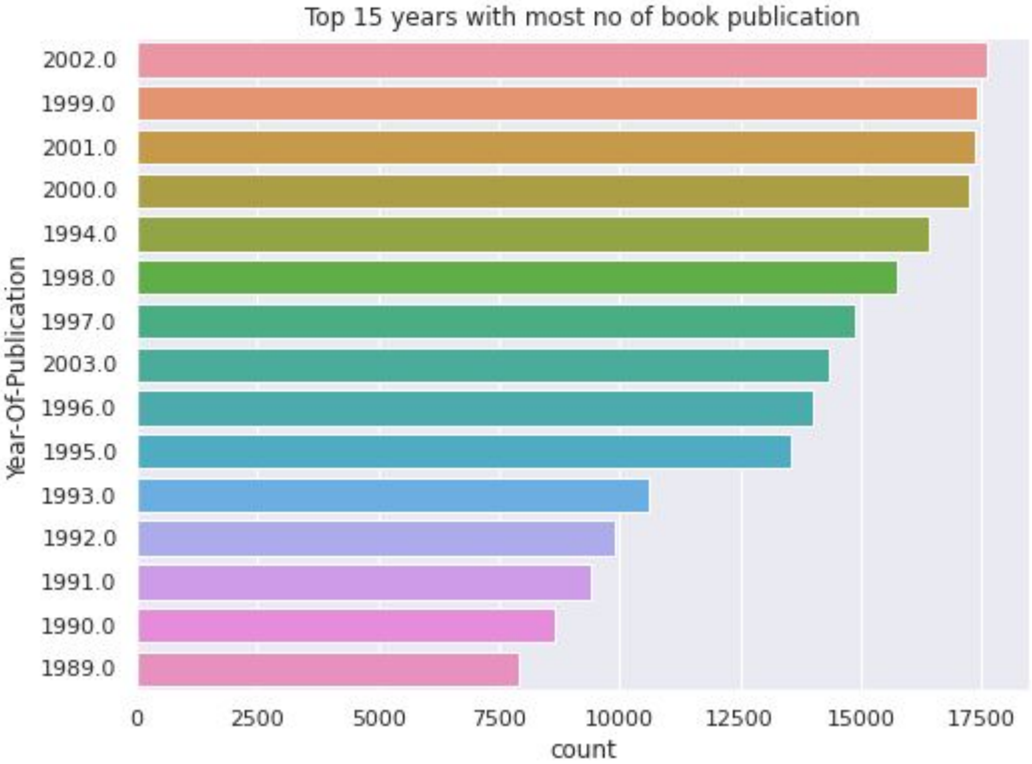
15 Most Read books

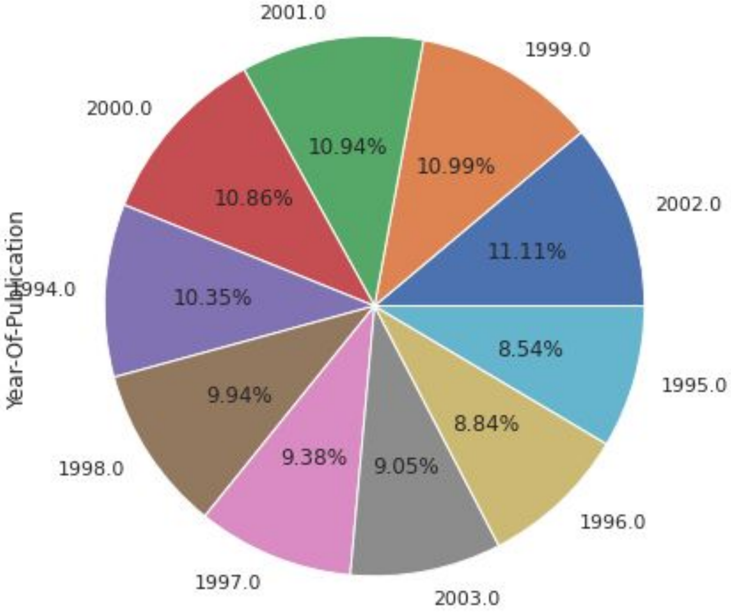


Author with most no of books

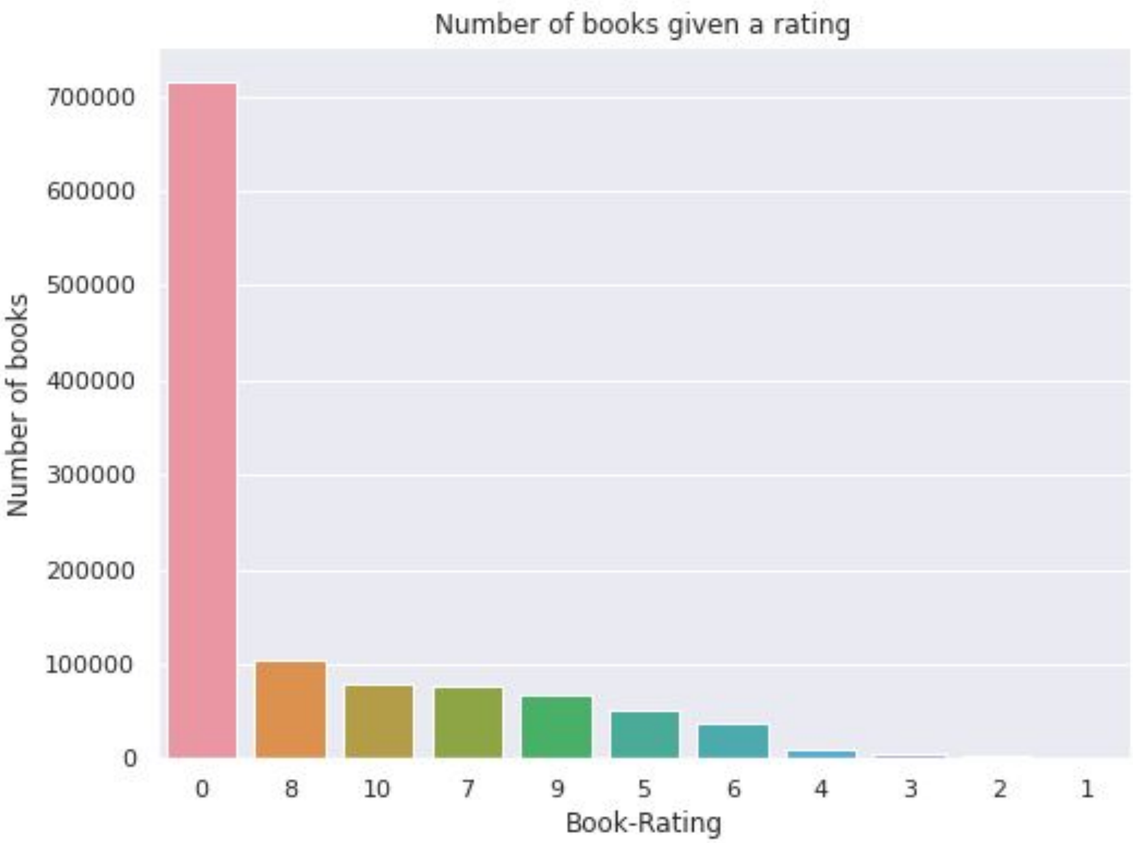


Year in which most books are published

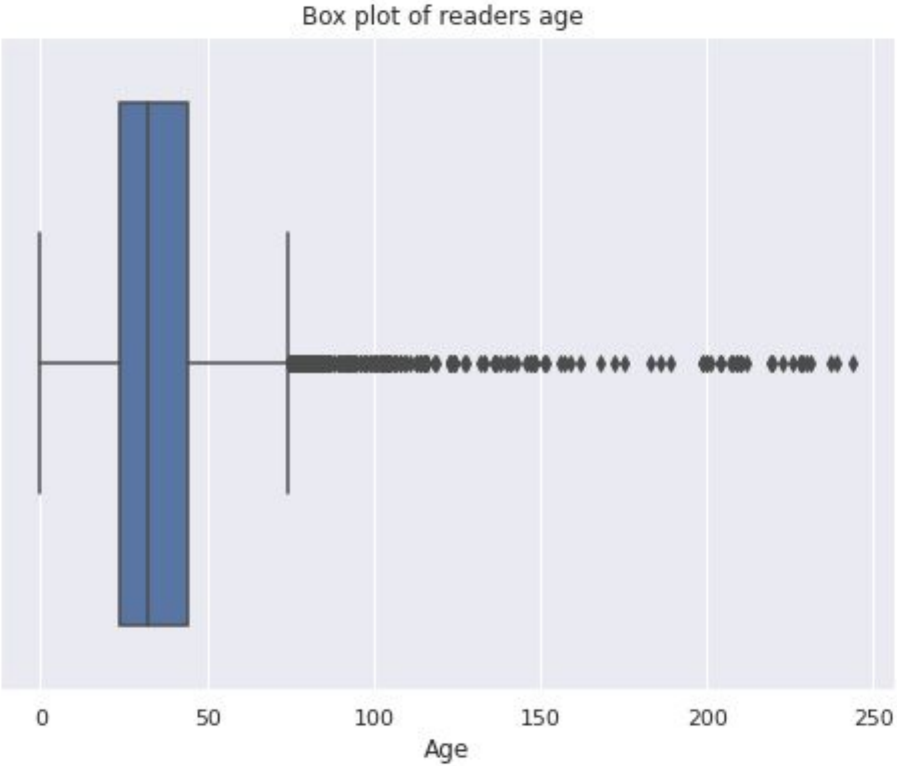


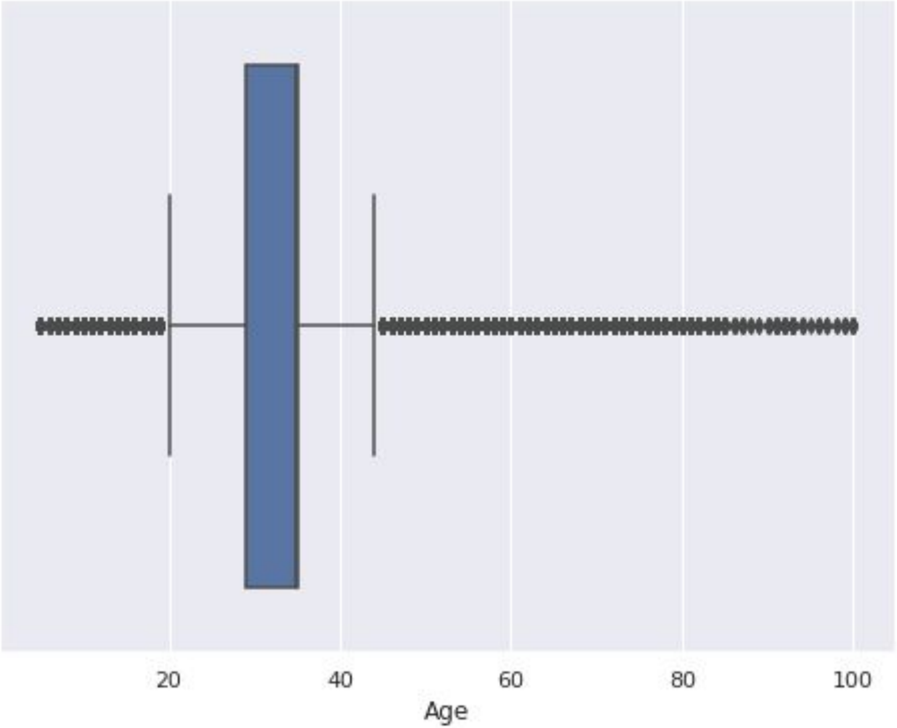


Rating of books



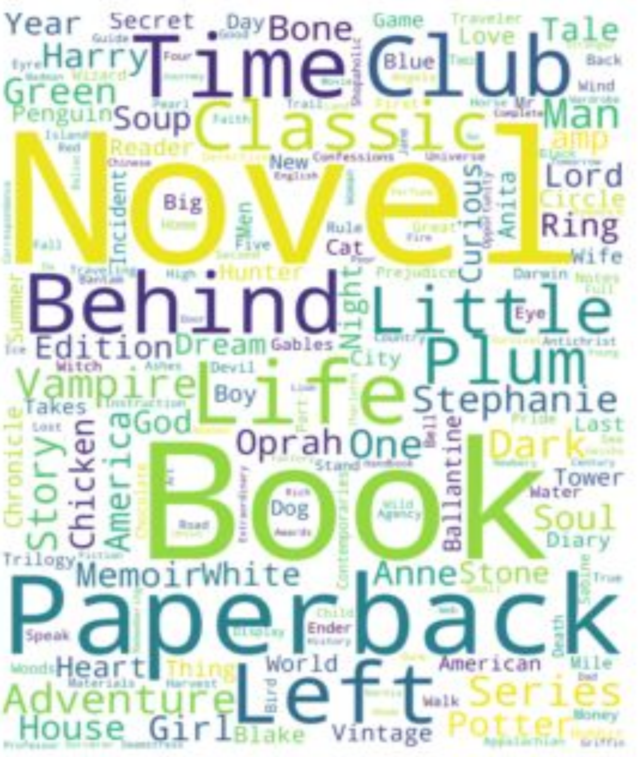
15 Most Served Cuisines



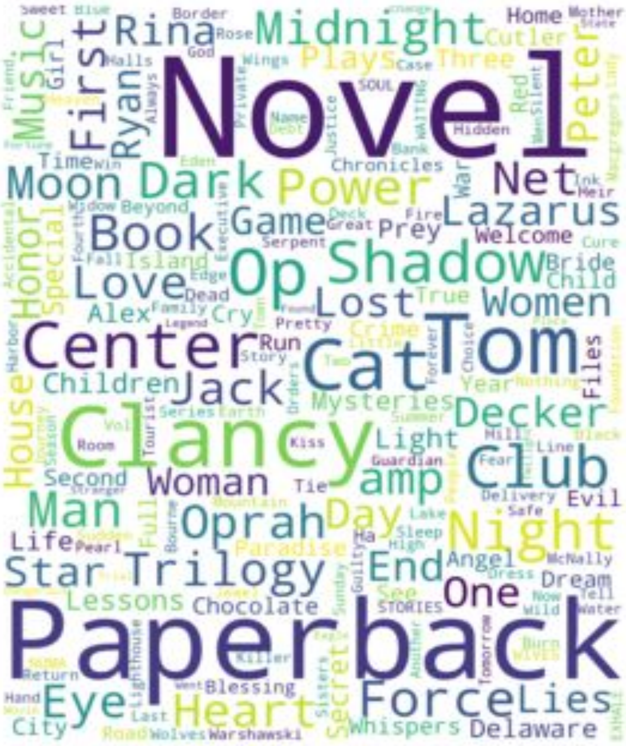


Word cloud for book titles

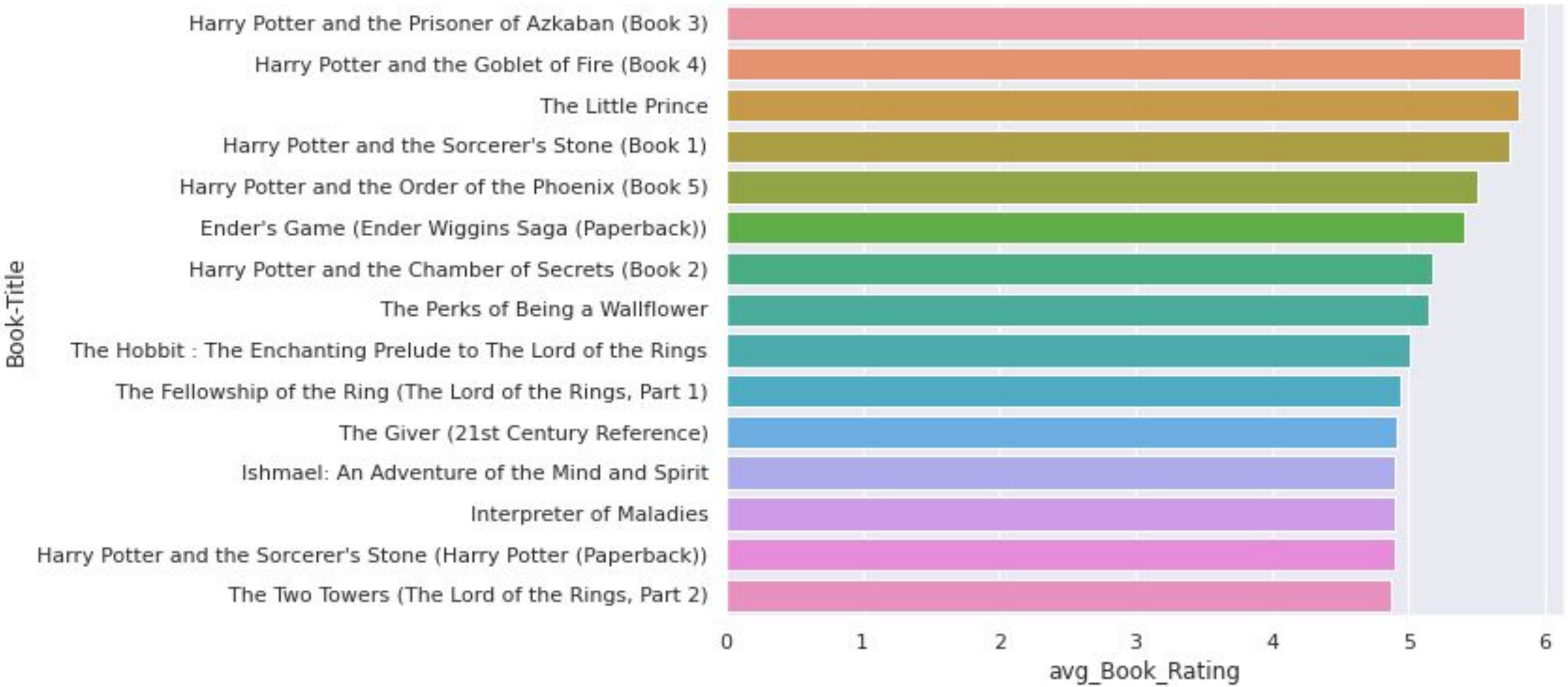
Top 500 books by rating



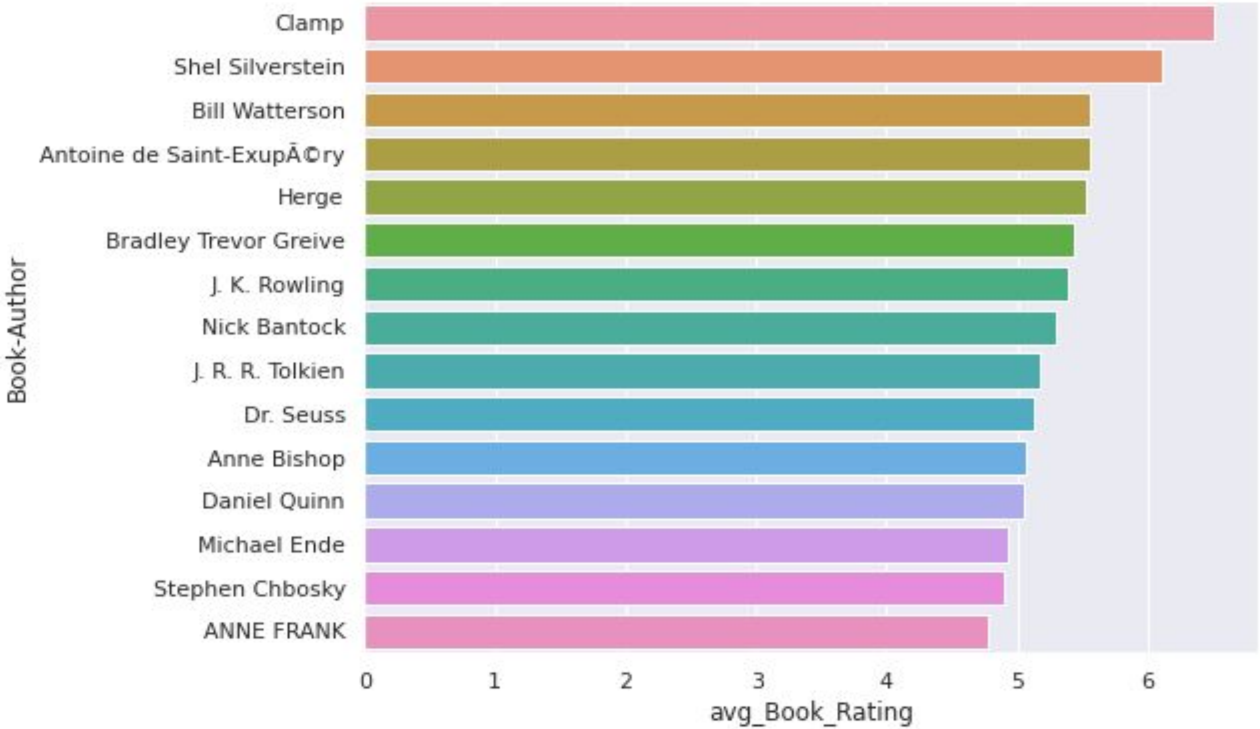
Bottom 500 books by rating



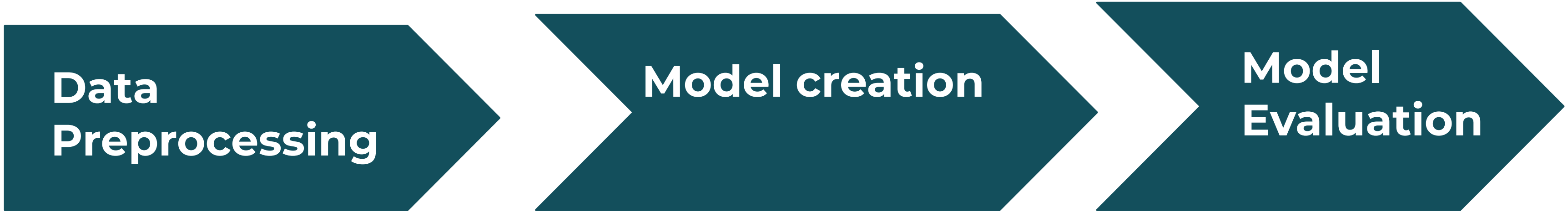
Top 15 highest rated books



Top 15 Author



Modeling Steps



Data Preprocessing

Model creation

Model Evaluation

- Feature selection
- Feature engineering
- Feature Extraction
- Train test data split(75%-25%)

- Nearest Neighbours
- Model based collaborative filtering (SVD)
- Content based filtering

- Top-N accuracy metrics

Nearest neighbours

NN is a machine learning algorithm to find clusters of similar users based on common book ratings, and make predictions using the average rating of top-n nearest neighbors.

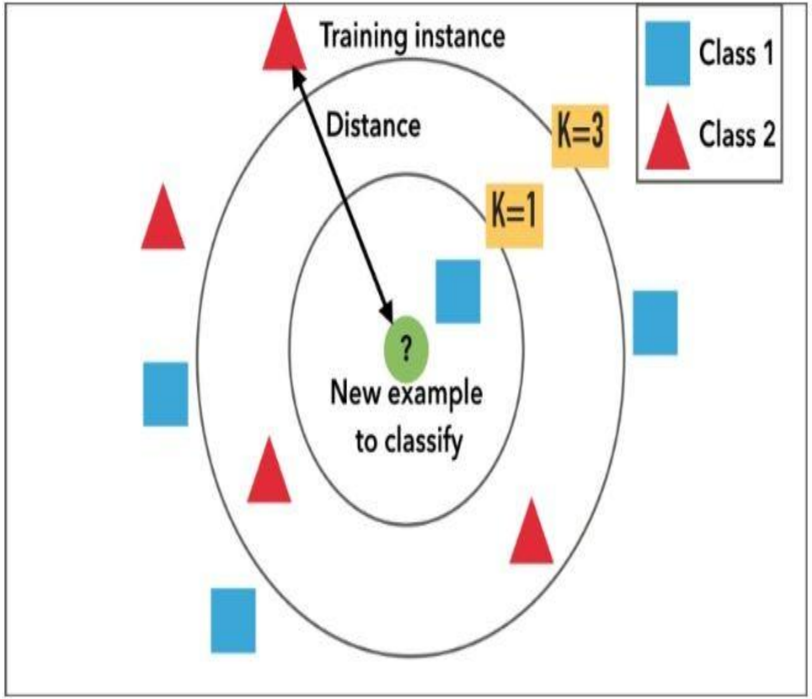
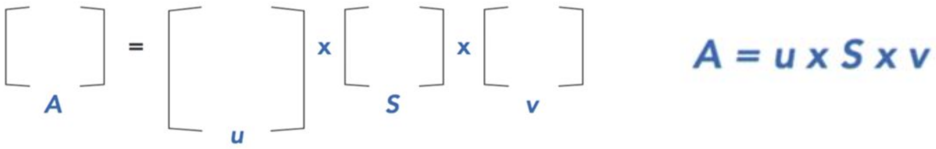


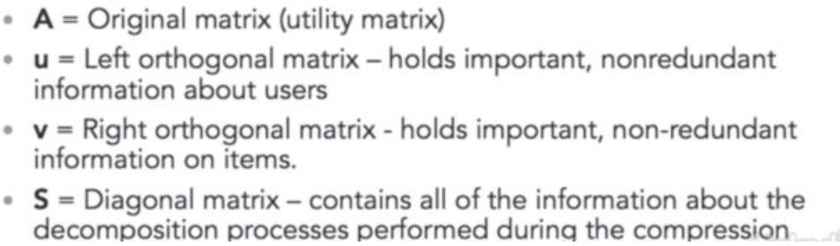
Illustration of how KNN makes classification about new sample

Collaborative Filtering Using Singular Value Decomposition (SVD)

The Singular-Value Decomposition, is a matrix decomposition method for reducing a matrix to its constituent parts in order to make certain subsequent matrix calculations simpler. It provides another way to factorize a matrix, into singular vectors and singular values.



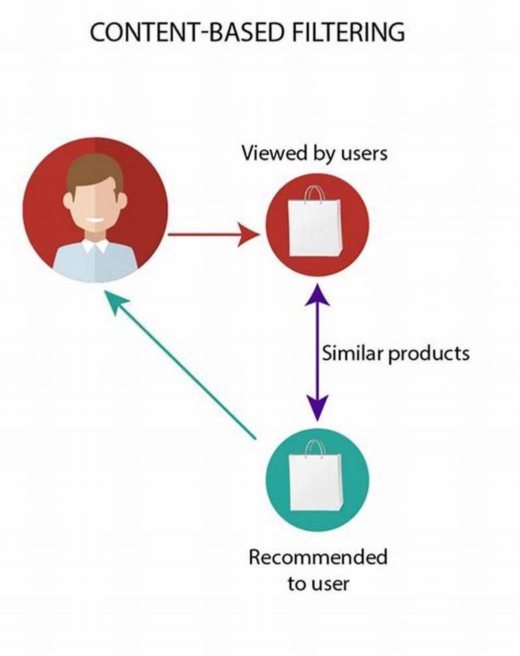
$A = u \times S \times v$



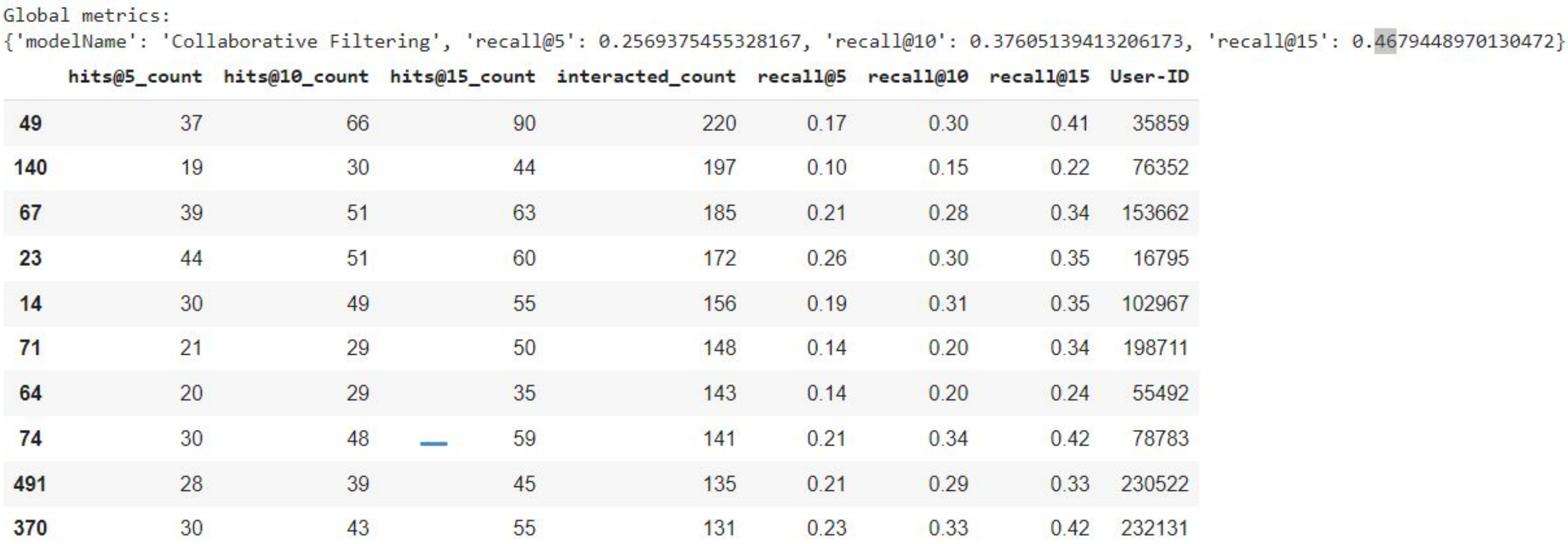
- **A** = Original matrix (utility matrix)
- **u** = Left orthogonal matrix – holds important, nonredundant information about users
- **v** = Right orthogonal matrix - holds important, non-redundant information on items.
- **S** = Diagonal matrix – contains all of the information about the decomposition processes performed during the compression

Content based filtering

Content-Based Recommendations systems look for similarity before recommending something. For eg whenever we are looking for a movie or web series on Netflix, we get the same genre movie recommended by Netflix. The similarity of different movies is computed to the one you are currently watching and all the similar movies are recommended to us. In the case of e-commerce website similarity in terms of products is calculated. Considering I am looking for a MacBook then the website will look for all similar products that are similar to MacBook and straight away will recommend us.



Score Matrix



```
Global metrics:
{'modelName': 'Collaborative Filtering', 'recall@5': 0.2569375455328167, 'recall@10': 0.37605139413206173, 'recall@15': 0.4679448970130472}
```

	hits@5_count	hits@10_count	hits@15_count	interacted_count	recall@5	recall@10	recall@15	User-ID
49	37	66	90	220	0.17	0.30	0.41	35859
140	19	30	44	197	0.10	0.15	0.22	76352
67	39	51	63	185	0.21	0.28	0.34	153662
23	44	51	60	172	0.26	0.30	0.35	16795
14	30	49	55	156	0.19	0.31	0.35	102967
71	21	29	50	148	0.14	0.20	0.34	198711
64	20	29	35	143	0.14	0.20	0.24	55492
74	30	48	59	141	0.21	0.34	0.42	78783
491	28	39	45	135	0.21	0.29	0.33	230522
370	30	43	55	131	0.23	0.33	0.42	232131

Challenges

- **High Volume of data which caused occasional crashing of system**
- **Elevating evaluation score for the models.**

Conclusion

- After comparing with content based and model based collaborative we came to the conclusion that Recall rate hit @ 15 around 47 for collaborative filtering.

Thank You