Capstone Project -4

Book Recommendation System

(individual)

Rajesh Kumar Patel

Introduction:

During the last few decades, with the rise of Youtube, Amazon, Netflix and many other such web services, recommender systems have taken more and more place in our lives. From e-commerce (suggest to buyers articles that could interest them) to online advertisement (suggest to users the right contents, matching their preferences), recommender systems are today unavoidable in our daily online journeys.

In a very general way, recommender systems are algorithms aimed at suggesting relevant items to users according to the interest, trend and popularity

Problem Statement:

The problem statement is to build a book recommendation system for users.

About dataset:

In this project we have used three dataset :

- Users Dataset - it contains the user information
- Books Dataset - it contains the book information
- Ratings Dataset - it contains the book rating information

Feature information in details

Dataset 1 - Users.csv (278858, 3)

- **User-ID** Unique user id
- **Location** location of the user
- **Age** age of user

Dataset 2 - Books.csv (271360, 8)

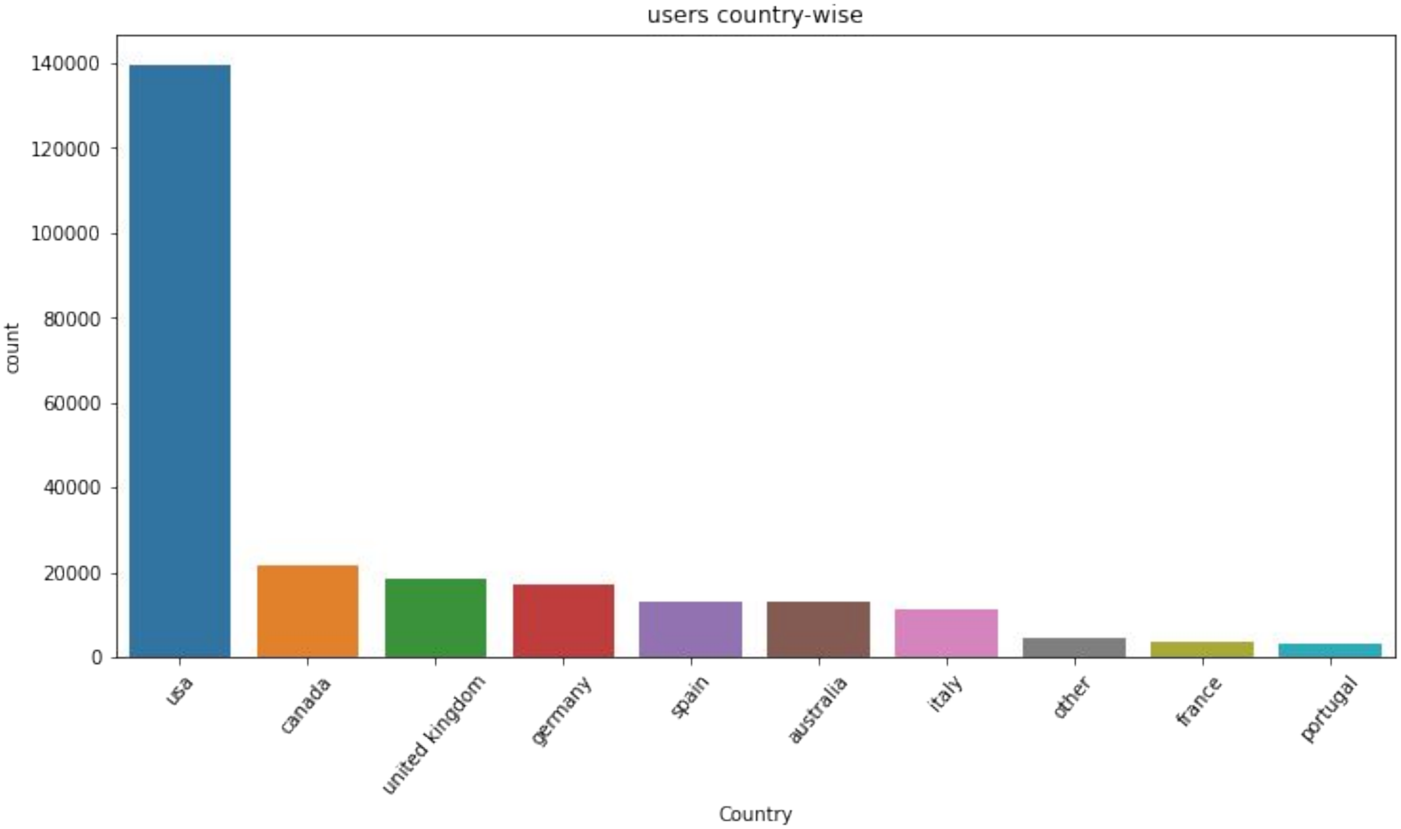
- **ISBN** Book ID
- **Book-Title** Book Name
- **Book-Author** Book Author Name
- **Year-Of-Publication** year of publication date
- **Publisher** publisher of the book
- **Image-URL-S** small image of the book , amazon link
- **Image-URL-M** medium size image of the book , amazon link
- **Image-URL-L** large image size of the book , amazon link

Dataset 3 - Rating.csv (1149780, 3)

- **User-ID** Unique user id
- **ISBN** Book ID
- **Book-Rating** Rating given by user

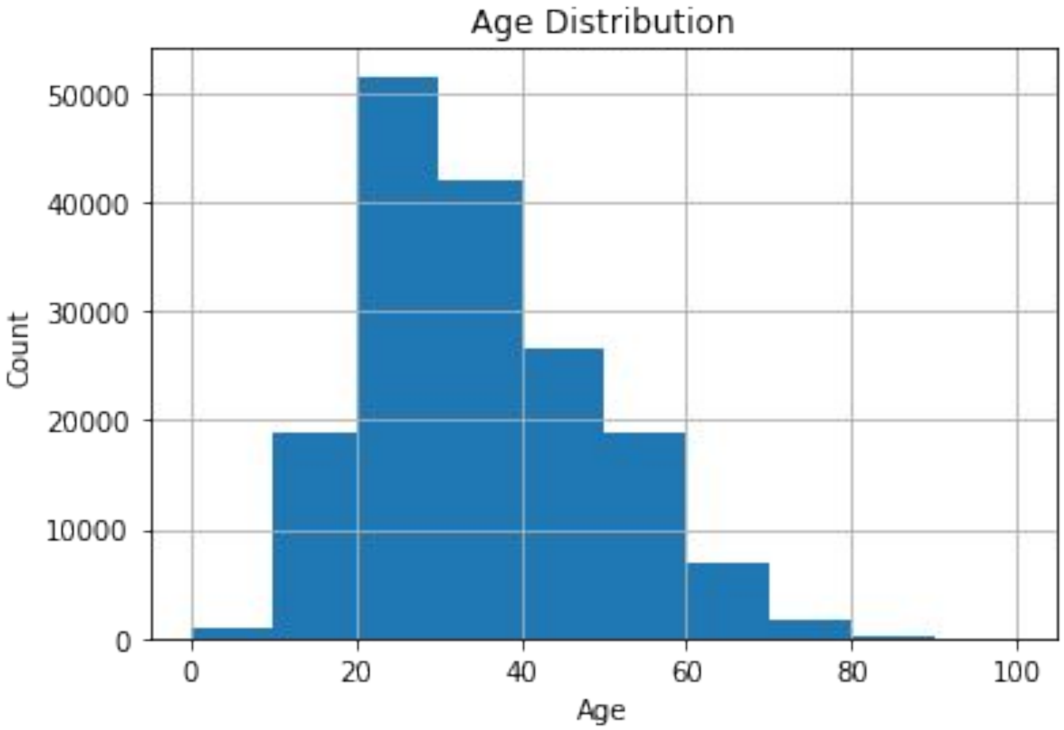
Exploratory Data Analysis

Users Dataset
Location feature



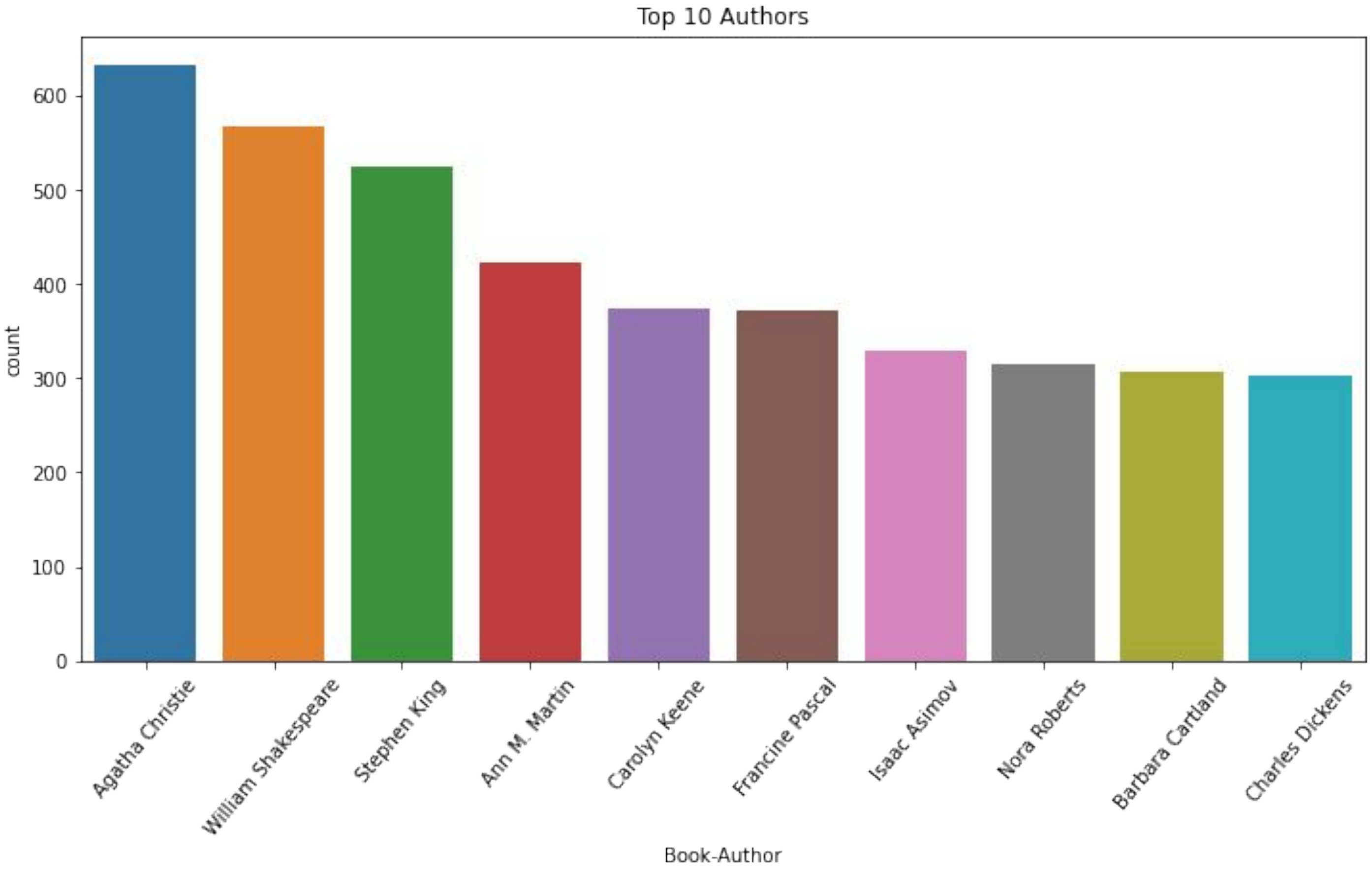
Most of the users belongs to the USA Location

Users Dataset
Age feature



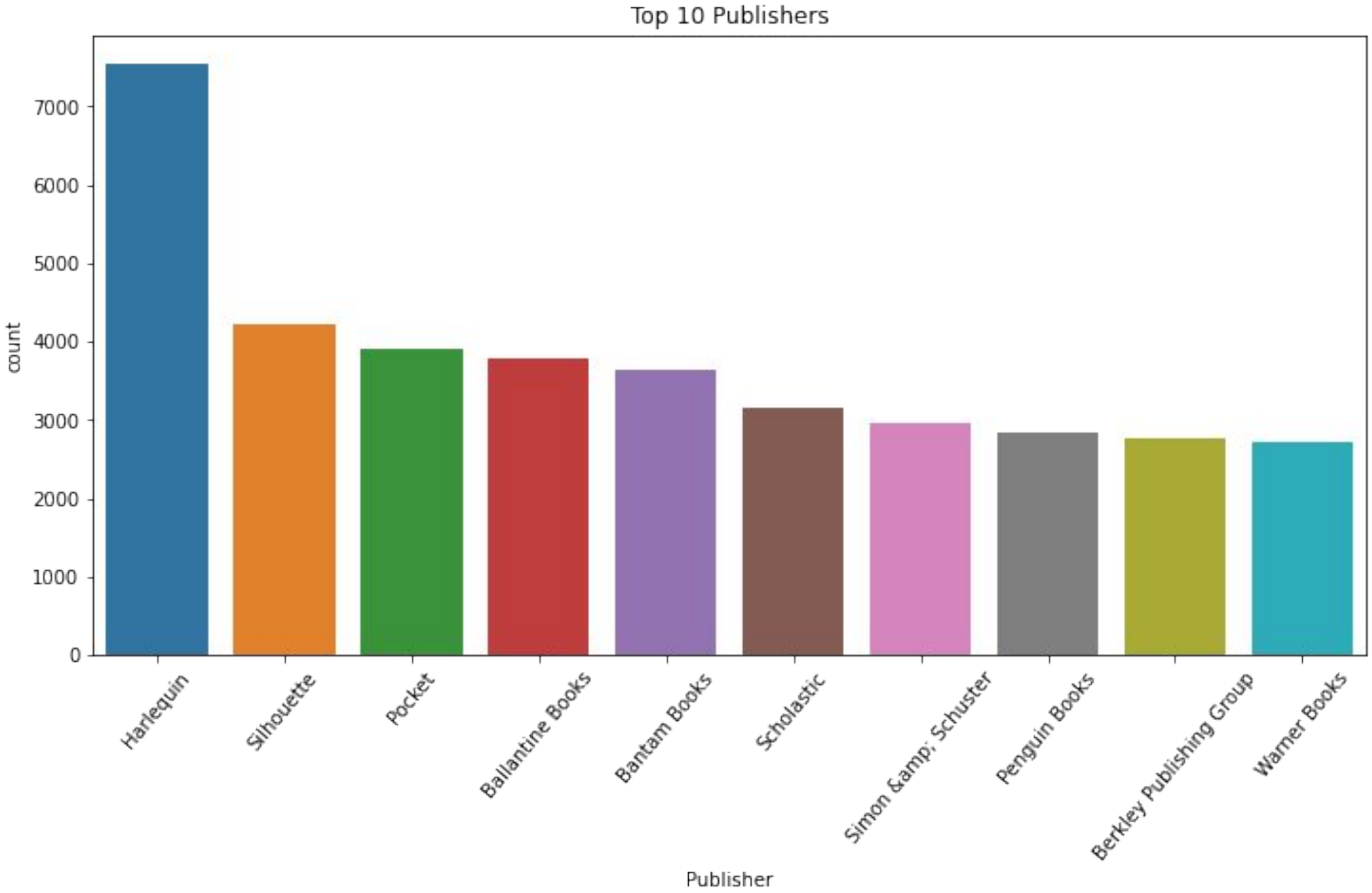
Most of the user’s are between 20 to 30 years.

Books Dataset Book-Author feature



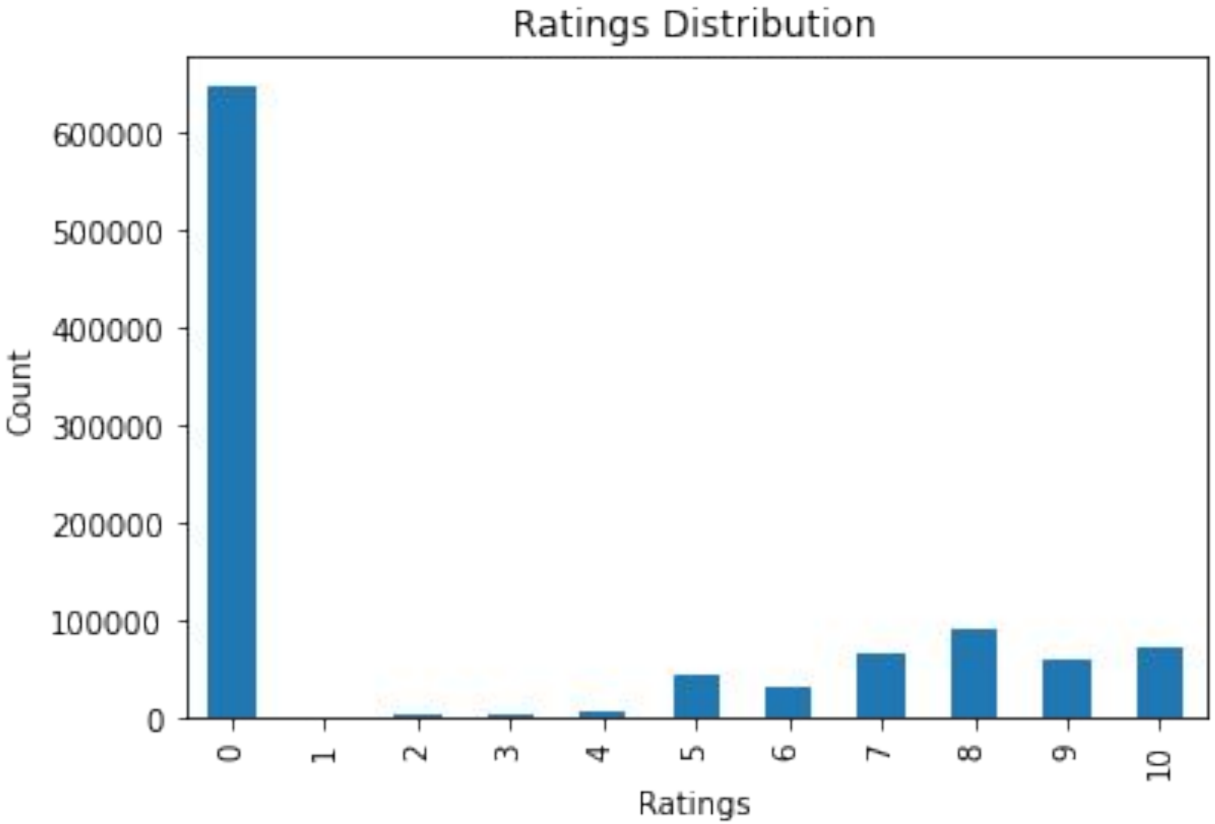
Agatha Christie is the top Author according to more books written.

Books Dataset Publisher feature



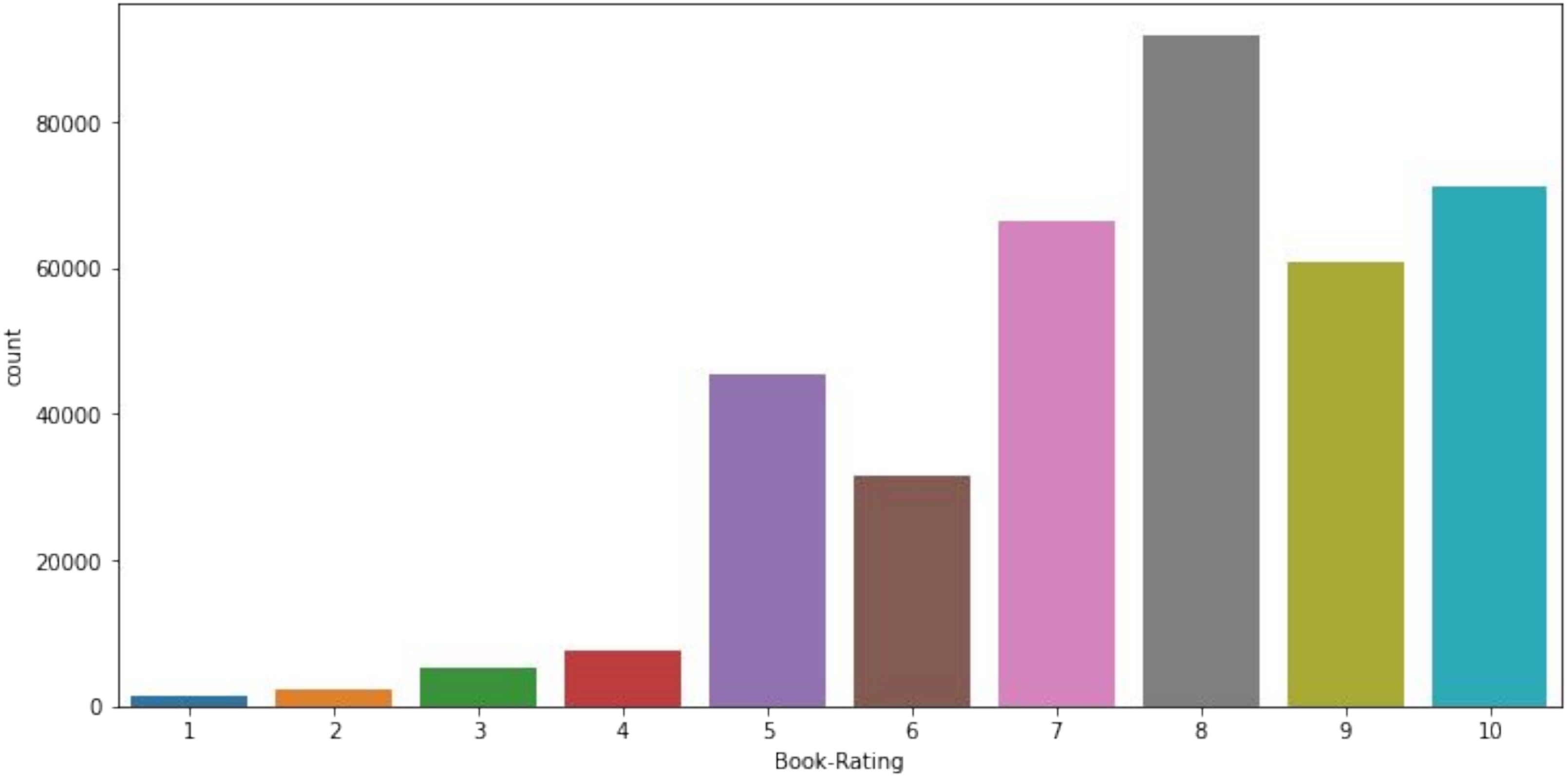
Harlequin is the top Publisher according to Publish more books.

Ratings Dataset
Book-Rating
feature



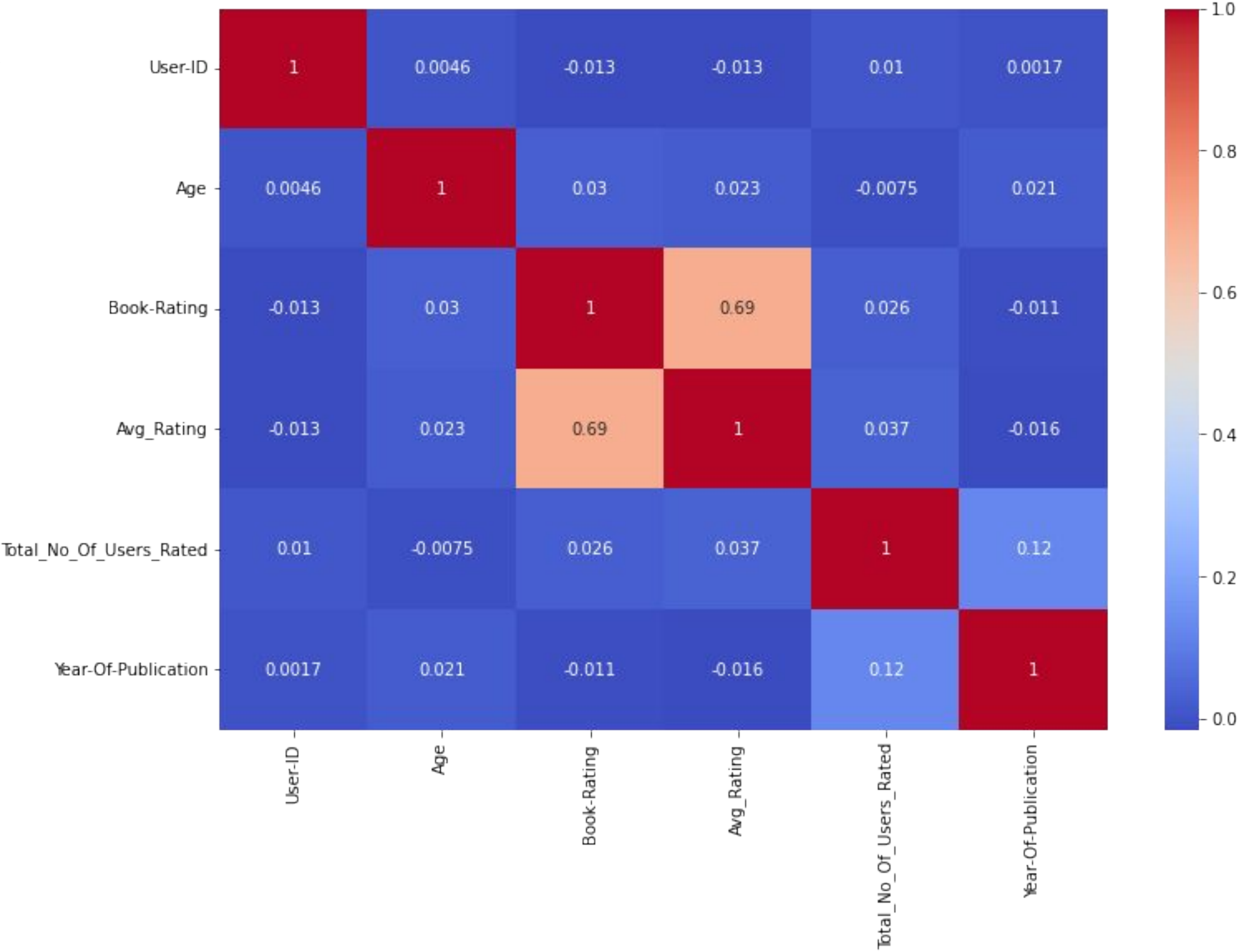
The ratings are very unevenly distributed, and the vast majority of ratings are 0
.

Ratings Dataset Book-Rating feature (non-zero ratings)



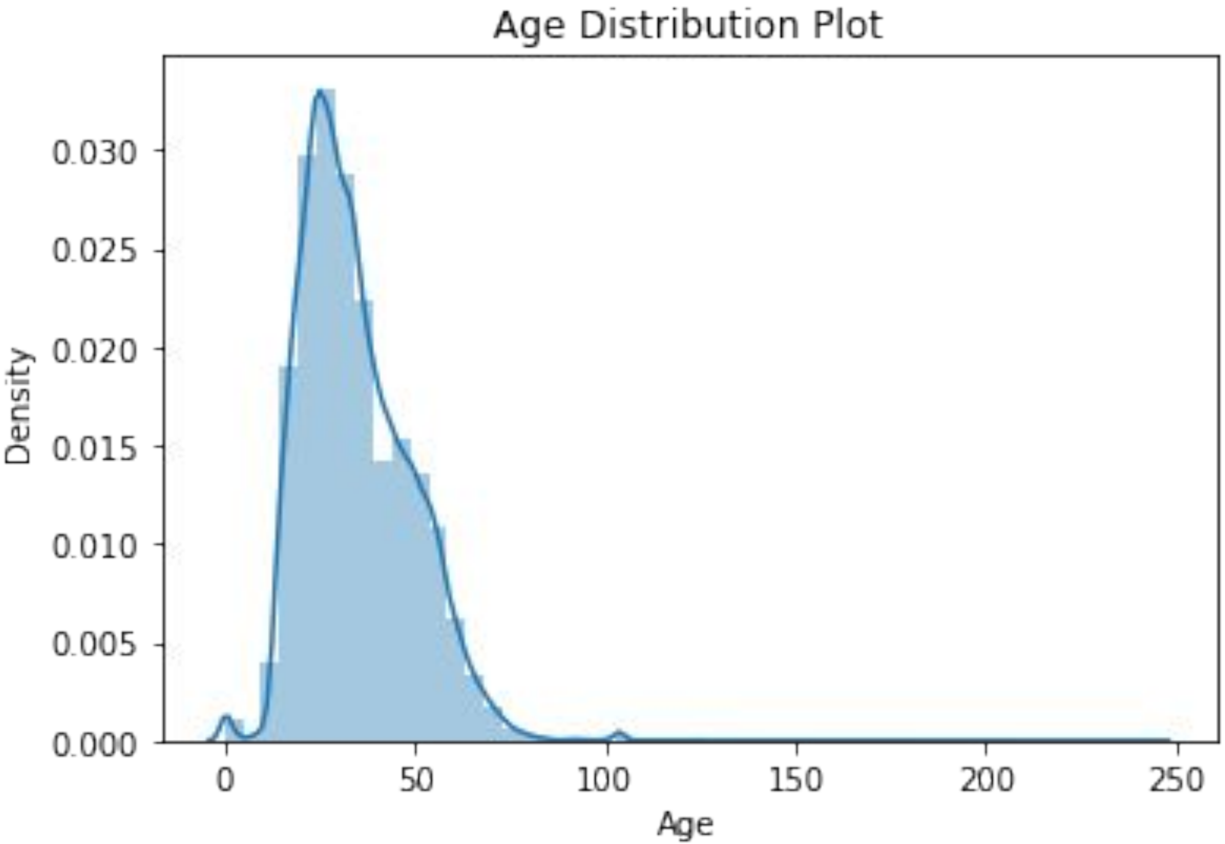
Most of the books having ratings value is 8.

Correlation matrix of Merged Dataset



	User-ID	Age	Book-Rating	Avg_Rating	Total_No_Of_Users_Rated	Year-Of-Publication
User-ID	1	0.0046	-0.013	-0.013	0.01	0.0017
Age	0.0046	1	0.03	0.023	-0.0075	0.021
Book-Rating	-0.013	0.03	1	0.69	0.026	-0.011
Avg_Rating	-0.013	0.023	0.69	1	0.037	-0.016
Total_No_Of_Users_Rated	0.01	-0.0075	0.026	0.037	1	0.12
Year-Of-Publication	0.0017	0.021	-0.011	-0.016	0.12	1

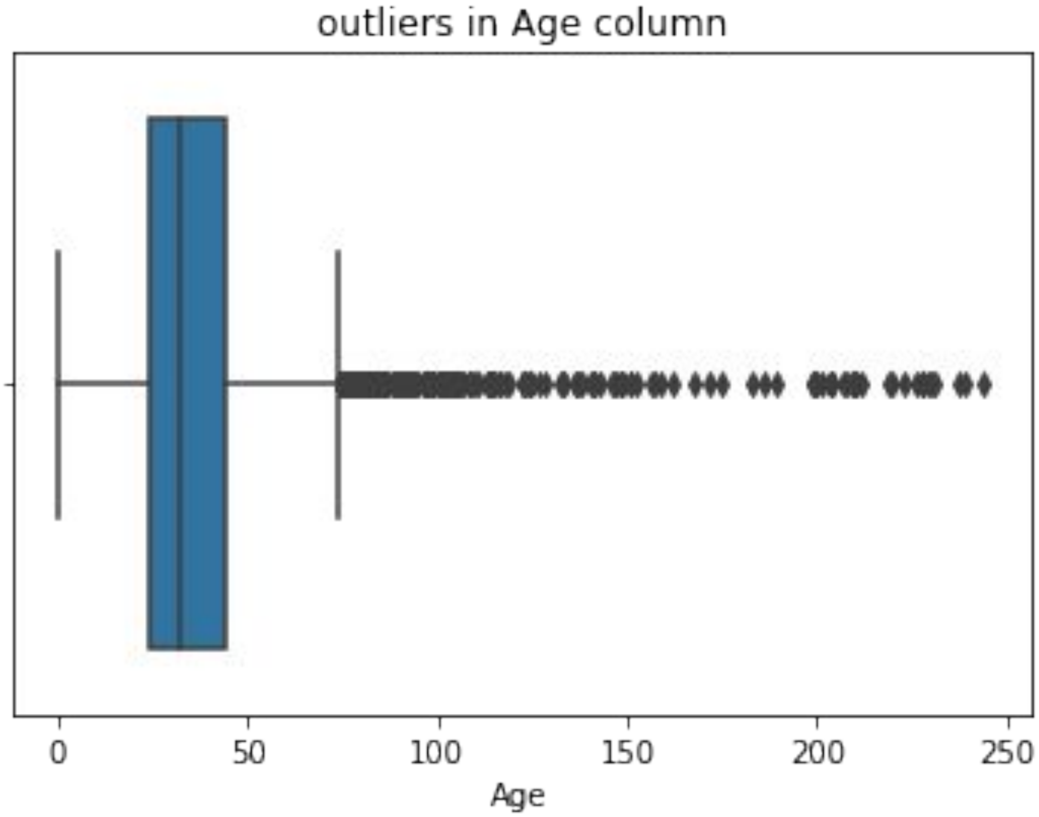
Feature Engineering



Age column have 39.7 % Null Value,

we fill NaN value in Age column by median with country-wise

Outliers present in Age column



Age value below 5 and above 100 do not make much sense for our book rating.

We have selected the only important feature that is more useful in our analysis and created some new feature like:

- Country
- Avg_Rating
- Total_No_Of_Users_Rated

Recommendation Model

used in our project

1. Popularity-Based Recommendation System
2. Model-Based Collaborative filtering
3. Item-Based Collaborative filtering
4. User-Based Collaborative filtering

1. Popularity-Based Recommendation System

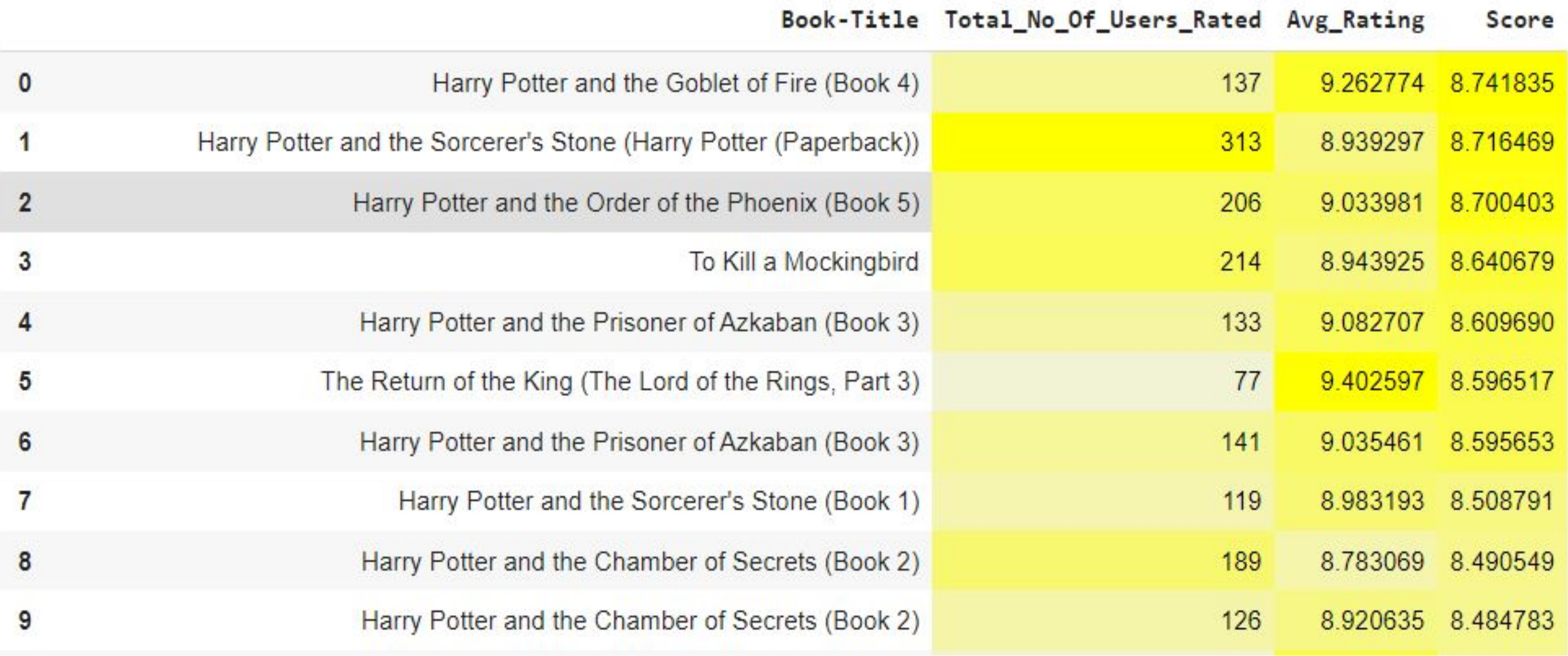
It is a type of recommendation system which works on the principle of popularity and or anything which is in trend. These systems check about the book which are in trend or are most popular among the users and directly recommend those.

Book weighted average formula:

Weighted_Rating(WR)= [vR/(v+m)] + [mC/(v+m)]

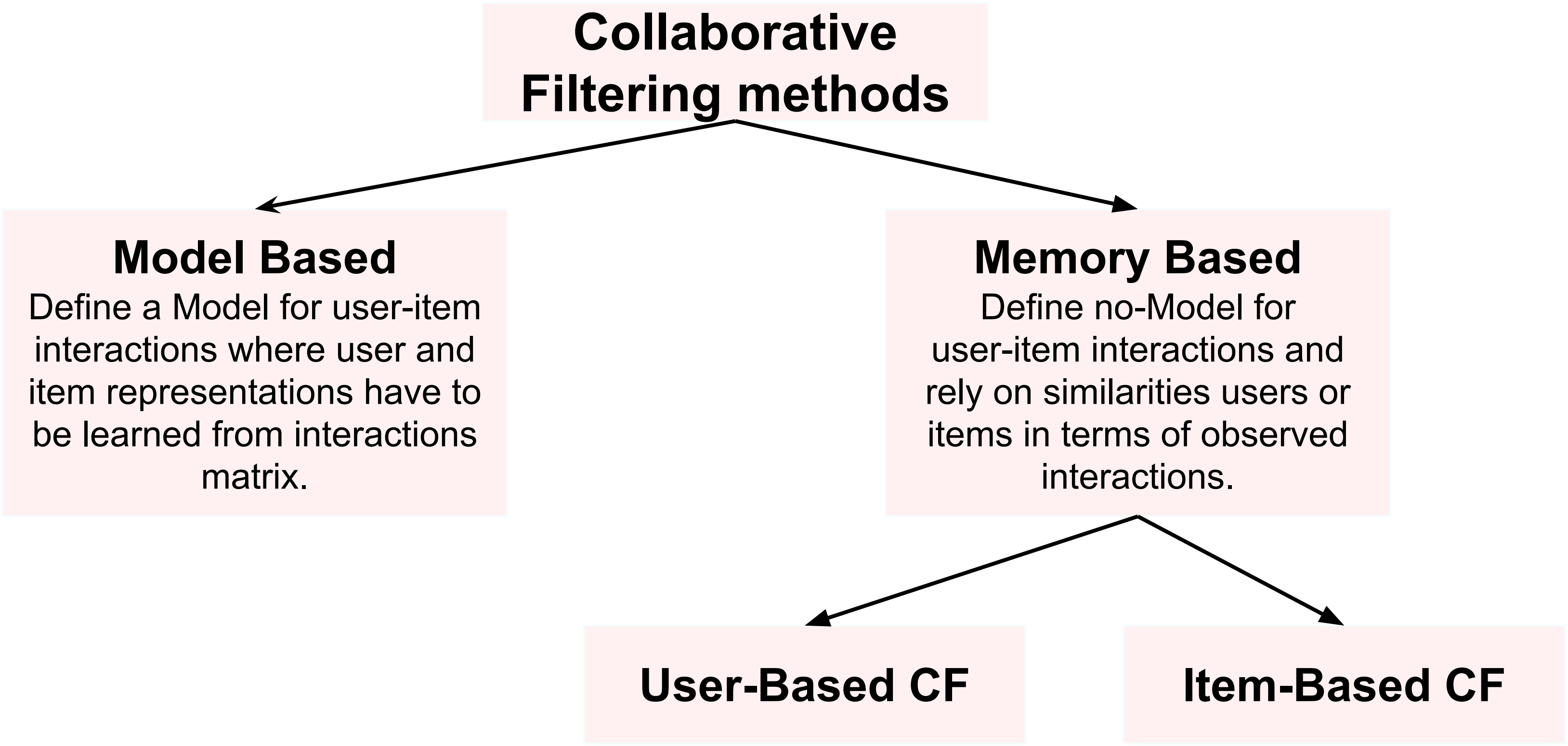
Where:
v is the number of rating for the books;
m is the minimum rating required to be listed in the chart;
R is the average rating of the book; and
C is the mean rating across the whole ratings.

Popularity-Based Recommendation System



	Book-Title	Total_No_Of_Users_Rated	Avg_Rating	Score
0	Harry Potter and the Goblet of Fire (Book 4)	137	9.262774	8.741835
1	Harry Potter and the Sorcerer's Stone (Harry Potter (Paperback))	313	8.939297	8.716469
2	Harry Potter and the Order of the Phoenix (Book 5)	206	9.033981	8.700403
3	To Kill a Mockingbird	214	8.943925	8.640679
4	Harry Potter and the Prisoner of Azkaban (Book 3)	133	9.082707	8.609690
5	The Return of the King (The Lord of the Rings, Part 3)	77	9.402597	8.596517
6	Harry Potter and the Prisoner of Azkaban (Book 3)	141	9.035461	8.595653
7	Harry Potter and the Sorcerer's Stone (Book 1)	119	8.983193	8.508791
8	Harry Potter and the Chamber of Secrets (Book 2)	189	8.783069	8.490549
9	Harry Potter and the Chamber of Secrets (Book 2)	126	8.920635	8.484783

This recommendation result for all user



Collaborative Filtering methods

Model Based

Define a Model for user-item interactions where user and item representations have to be learned from interactions matrix.

Memory Based

Define no-Model for user-item interactions and rely on similarities users or items in terms of observed interactions.

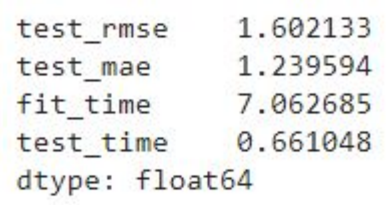
User-Based CF

Item-Based CF

2. Model-Based Collaborative filtering

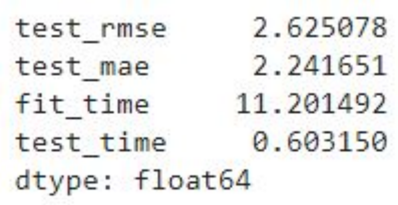
we have used two type of model: Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF)

SVD Model:



```
test_rmse     1.602133
test_mae      1.239594
fit_time      7.062685
test_time     0.661048
dtype: float64
```

NMF Model:

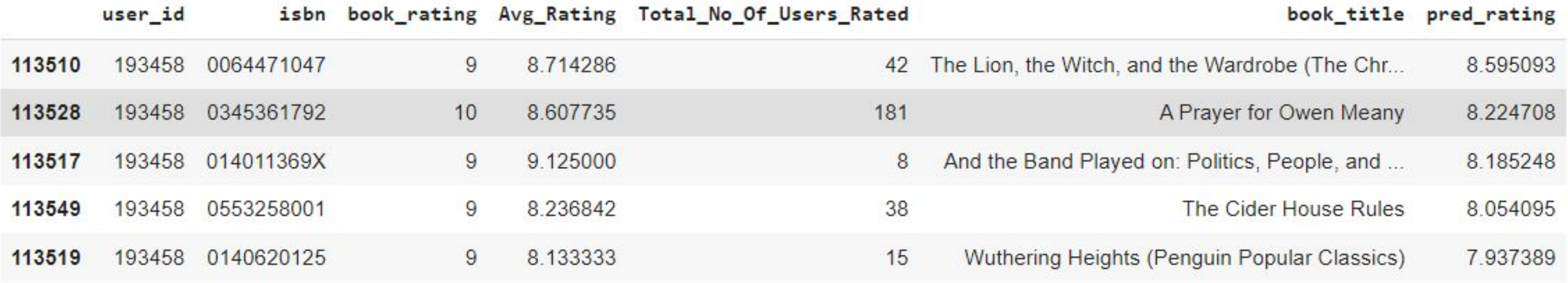


```
test_rmse      2.625078
test_mae       2.241651
fit_time      11.201492
test_time      0.603150
dtype: float64
```

It's clear that for the given dataset much better results can be obtained with SVD approach - both in terms of accuracy and training / testing time.

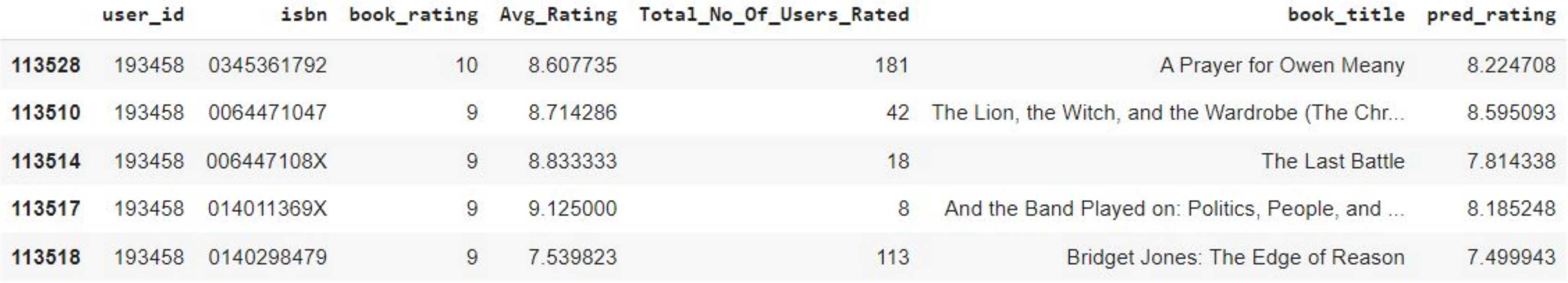
SVD Model result:

predicted top rating books



	user_id	isbn	book_rating	Avg_Rating	Total_No_Of_Users_Rated	book_title	pred_rating
113510	193458	0064471047	9	8.714286	42	The Lion, the Witch, and the Wardrobe (The Chr...	8.595093
113528	193458	0345361792	10	8.607735	181	A Prayer for Owen Meany	8.224708
113517	193458	014011369X	9	9.125000	8	And the Band Played on: Politics, People, and ...	8.185248
113549	193458	0553258001	9	8.236842	38	The Cider House Rules	8.054095
113519	193458	0140620125	9	8.133333	15	Wuthering Heights (Penguin Popular Classics)	7.937389

Actual top rating books



	user_id	isbn	book_rating	Avg_Rating	Total_No_Of_Users_Rated	book_title	pred_rating
113528	193458	0345361792	10	8.607735	181	A Prayer for Owen Meany	8.224708
113510	193458	0064471047	9	8.714286	42	The Lion, the Witch, and the Wardrobe (The Chr...	8.595093
113514	193458	006447108X	9	8.833333	18	The Last Battle	7.814338
113517	193458	014011369X	9	9.125000	8	And the Band Played on: Politics, People, and ...	8.185248
113518	193458	0140298479	9	7.539823	113	Bridget Jones: The Edge of Reason	7.499943

3. Item-Based Collaborative filtering

To make a new recommendation to a user, the idea of the Item-Based method is to find items similar to the ones the user already “positively” interacted with.

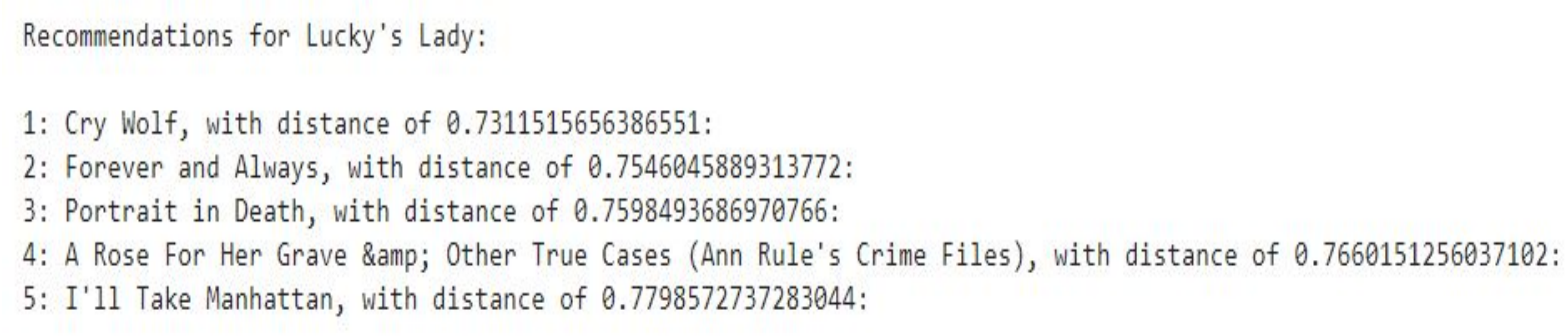
Two items are considered to be similar if most of the users that have interacted with both of them did it in a similar way.

This method is said to be “item-centred” as it represents items based on interactions users had with them and evaluates distances between those items.

For the scope of our project, we used the K-Nearest Neighbours algorithm.

Item-Based Collaborative filtering

Making books Recommendations



```
Recommendations for Lucky's Lady:

1: Cry Wolf, with distance of 0.7311515656386551:
2: Forever and Always, with distance of 0.7546045889313772:
3: Portrait in Death, with distance of 0.7598493686970766:
4: A Rose For Her Grave &amp; Other True Cases (Ann Rule's Crime Files), with distance of 0.7660151256037102:
5: I'll Take Manhattan, with distance of 0.7798572737283044:
```

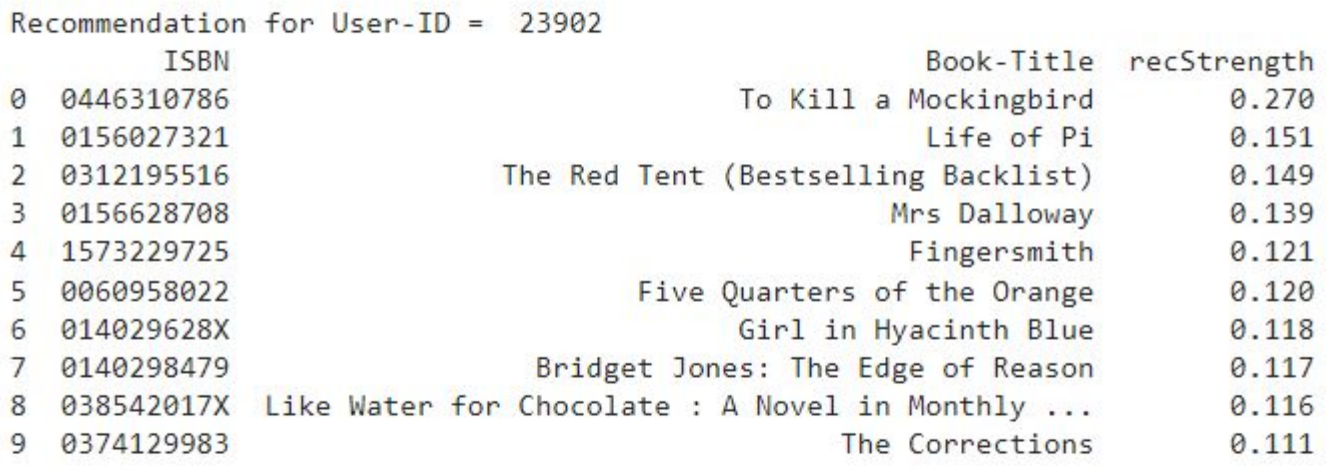
4. User-Based Collaborative filtering

In order to make a new recommendation to a user, the User-Based method roughly tries to identify users with the most similar “interactions profile” (nearest neighbours) in order to suggest items that are the most popular among these neighbours (and that are “new” to our user).

This method is said to be “user-centred” as it represents users based on their interactions with items and evaluates distances between users.

User-Based Collaborative filtering

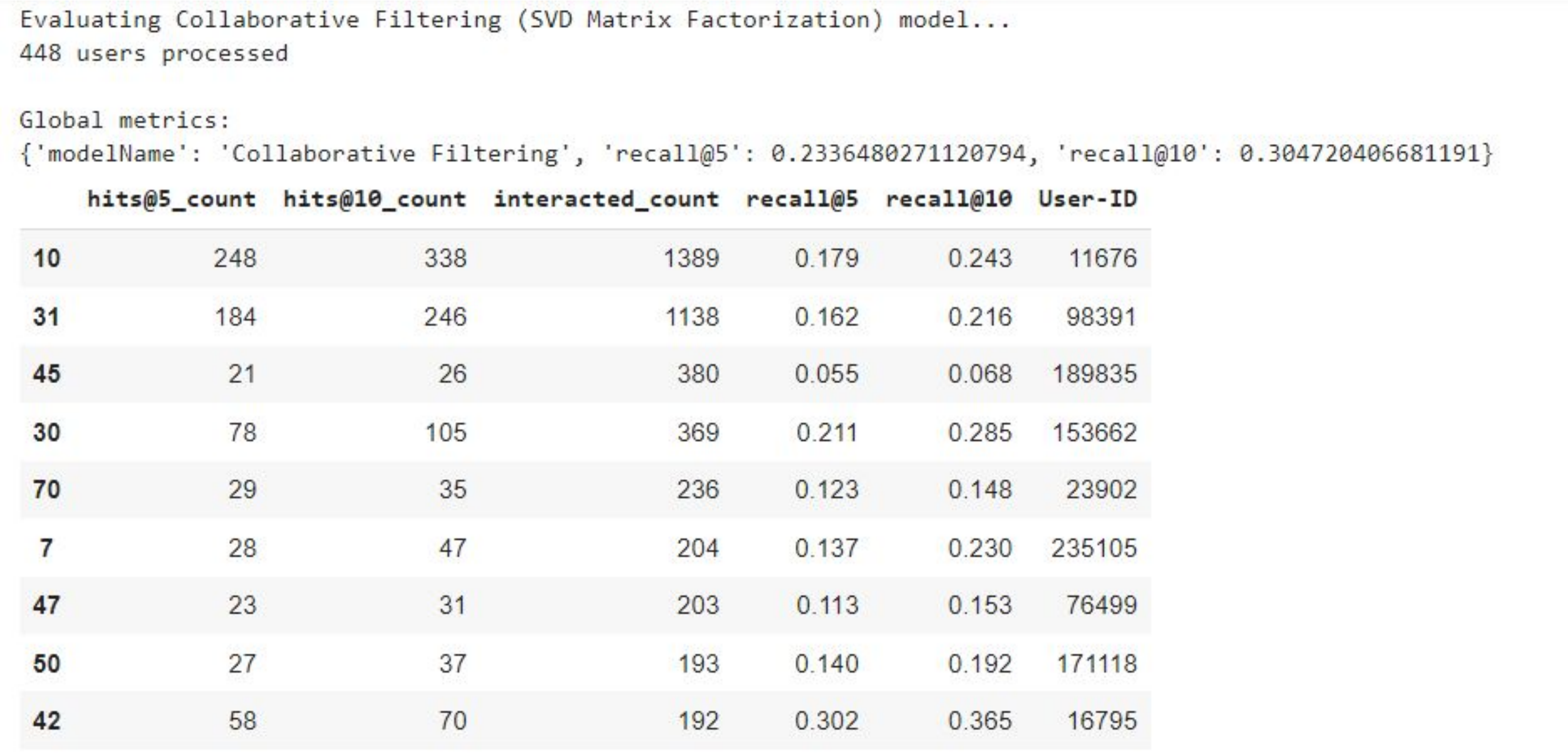
Making books Recommendations for User-ID 23902



```
Recommendation for User-ID =  23902
         ISBN                                        Book-Title  recStrength
0  0446310786                             To Kill a Mockingbird        0.270
1  0156027321                                       Life of Pi        0.151
2  0312195516              The Red Tent (Bestselling Backlist)        0.149
3  0156628708                                     Mrs Dalloway        0.139
4  1573229725                                     Fingersmith        0.121
5  0060958022                       Five Quarters of the Orange        0.120
6  014029628X                             Girl in Hyacinth Blue        0.118
7  0140298479                   Bridget Jones: The Edge of Reason        0.117
8  038542017X  Like Water for Chocolate : A Novel in Monthly ...        0.116
9  0374129983                                  The Corrections        0.111
```

User-Based Collaborative filtering

Model Result



```
Evaluating Collaborative Filtering (SVD Matrix Factorization) model...
448 users processed

Global metrics:
{'modelName': 'Collaborative Filtering', 'recall@5': 0.2336480271120794, 'recall@10': 0.304720406681191}
```

	hits@5_count	hits@10_count	interacted_count	recall@5	recall@10	User-ID
10	248	338	1389	0.179	0.243	11676
31	184	246	1138	0.162	0.216	98391
45	21	26	380	0.055	0.068	189835
30	78	105	369	0.211	0.285	153662
70	29	35	236	0.123	0.148	23902
7	28	47	204	0.137	0.230	235105
47	23	31	203	0.113	0.153	76499
50	27	37	193	0.140	0.192	171118
42	58	70	192	0.302	0.365	16795

Conclusion

Starting with loading the data so far we have done EDA, null values treatment, feature engineering, merged all dataset and then model building.

Popularity Based Recommender provides a general chart of recommended books to all the users. They are not sensitive to the interests and tastes of a particular user.

For modelling, it was observed that for model based collaborative filtering SVD technique worked way better than NMF with lower Mean Absolute Error (MAE).

Among the memory based collaborative filtering approaches, item-based CF performed better than user-based CF because of lower computation power.

Thank you!

Presented by: Rajesh Kumar Patel