Project

Project Title – Book Recommender System

By : Shaurabh Pandey

WHAT IS RECOMMENDER SYSTEM

A Recommender system refers to a system that is capable of predicting the future preference of a set of items for a user , and recommend the top items. In today's world, every customer is faced with multiple choices due to the prevalence of internet as compared to 30-35 years back when we used to shop in physical stores only where the choices are somewhat much limited. Netflix for example has an enormous movie library and so people had a hard time selecting the movie they actually want to watch and this is the kind of situation where the recommender system comes in , it learns from the past users data and provides recommendation to the users, without the user specifically searching for the item, the item is brought automatically by the system.

BOOKS RECOMMENDER SYSTEM

With the emergence of a number of e commerce platform around the world out of which some are dedicated solely to selling books and also thanks to new technologies in this field such as wireless electronic reading devices of which Kindle is the most popular example it has become extremely easy for the users to buy and read any kind of book from any author around the world .

BOOKS RECOMMENDER SYSTEM

This luxury of having almost a never ending list of options to choose from also brings with it a problem of many for the users, where they find it hard to decide on the one book they want to read right now. The Recommender system we are going to build will learn from the past user data basically the kind of books they read in the past and using all these knowledge it will recommend top n items that it thinks user will like to read in the future.

PROBLEM STATEMENT

Recommender systems aim to predict the “rating” or “preference” a user would give to an item

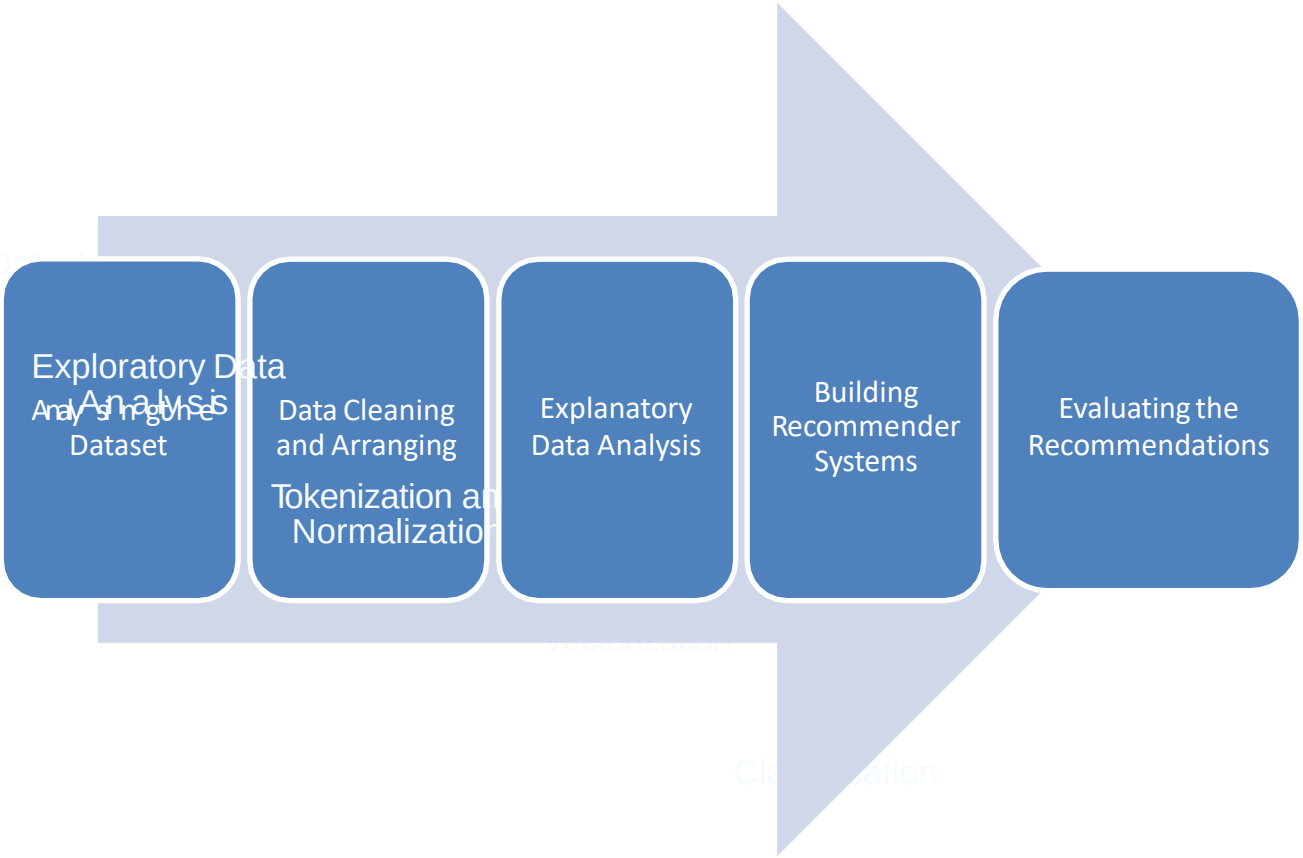
For this project we are provided with three files namely : books.csv , ratings.csv and users.csv . The books and the users dataset contained info on more than 2.7 lakh books and users respectively and the rating dataset contained around 12 lakh book rating information . Our job was to build a book recommender system using the info provided in these three datasets.

OBJECTIVE

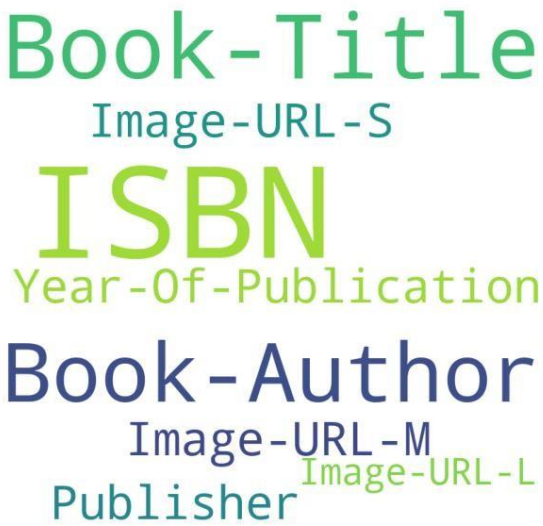
Analyze the Book-Crossing dataset files and get a general insight into the users taste and preference regarding the different books and authors in our dataset.

To build a book recommender system that can learn from all the past user data and generate top n recommendation for each individual user .

STEPS INVOLVED

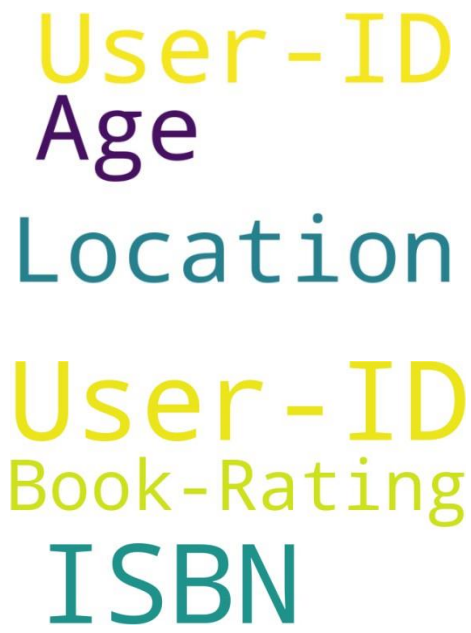


Defining the variables



- **Book-Title** : The title of the book
- **ISBN** : International Standard Book Number or ISBN numeric commercial book identifier that is intended to be unique.
- **Book-Author** : The author of the book
- **Year-Of-Publication** : The year in which this edition of the book was published
- **Publisher** : The Publishing house who published the book
- **Image-URL-L** : URL linking to large size cover image of the book
- **Image-URL-M** : URL linking to medium size cover image of the book
- **Image-URL-S** : URL linking to small size cover image of the book

Defining the variables



- **User-ID** : The anonymized user id of the users
- **Age** : The age of the user .
- **Location** : This contains the city, state and country of the user
- **Book-Rating** : This contains the book rating information , ratings are either explicit or implicit expressed by 0

Data Cleaning and Arranging

- In the books dataset we had some odd observations where the Year-Of-Publication column value was set to a string rather than a number .On a more closer look we found that the values in those observation were out of place by one position left and so we shifted the values in those observation by one place right

- Using Year-Of-Publication column we created a new column named decade to indicate the decade in which the book was published

- We changed the values of Book-Title , Book-Author and Publisher columns at first to lower case and then capitalized the first letter of all the words so to remove any case irregularities in them.

- Using the values in location column of users dataset we created three new columns named city , state and country

- Keeping the nature of our dataset in mind if the users age value in the users dataset was below 10 or above 100 , then in that case we replaced those values and the null values with a zero.

Data Cleaning and Arranging

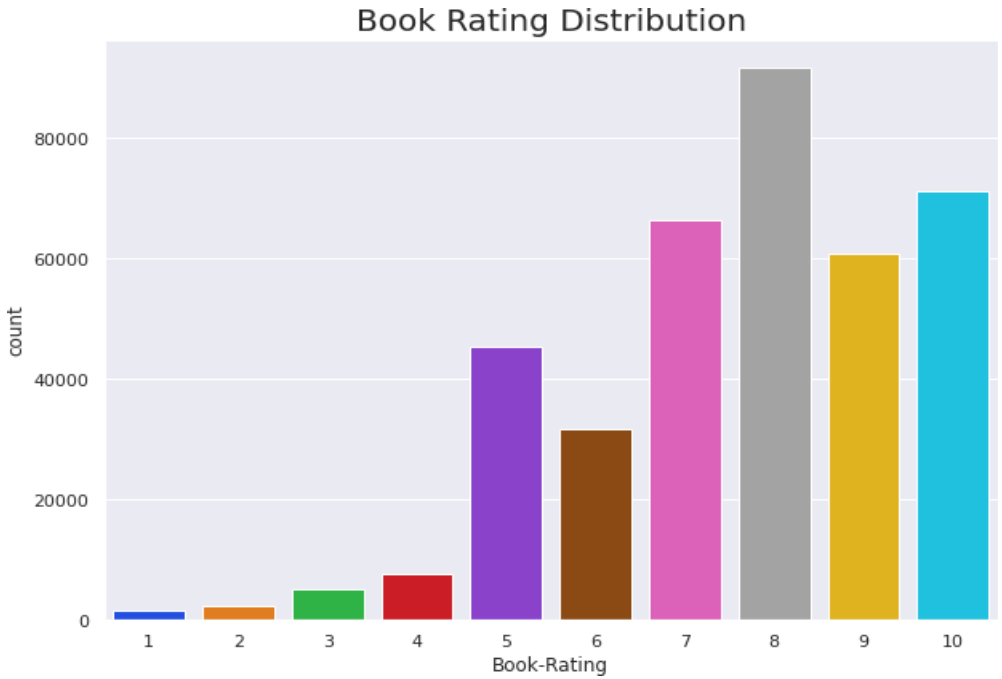
- Using the age values we created a new columns named Age-Group in which we categorized the age values into one of the 6 age groups we defined . Those 6 age groups were namely : Children , Teenager , Youth , Middle-Aged , Adulthood and Seniors .

- After merging the users , book and rating dataset into one we noticed that we had some duplicated values of rating for the same book by the same user but for a different ISBN number in that case we kept the record for the more recently published version .

- The ratings dataset contained both explicit ratings(expressed on a scale of 1-10) and implicit ratings given by zero , the recommender system we tried to build needed only explicit rating data so we dropped all the implicit ratings record

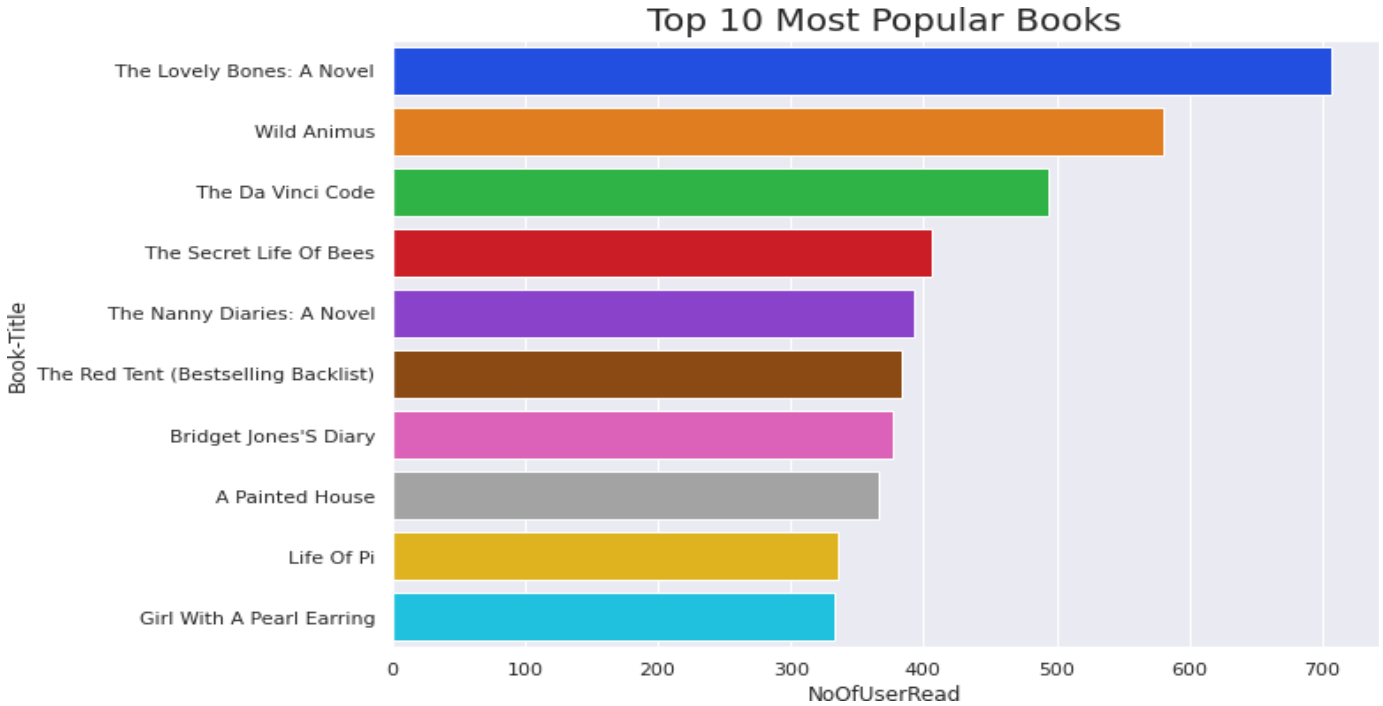
- To maintain statistical significance and so to generate meaningful recommendations , we only kept those books in our dataset with more than or equal to 15 user ratings and the same logic was applied in case of users as well where we dropped all those users from our dataset who have rated less than 15 books.

Book Rating Score Distribution



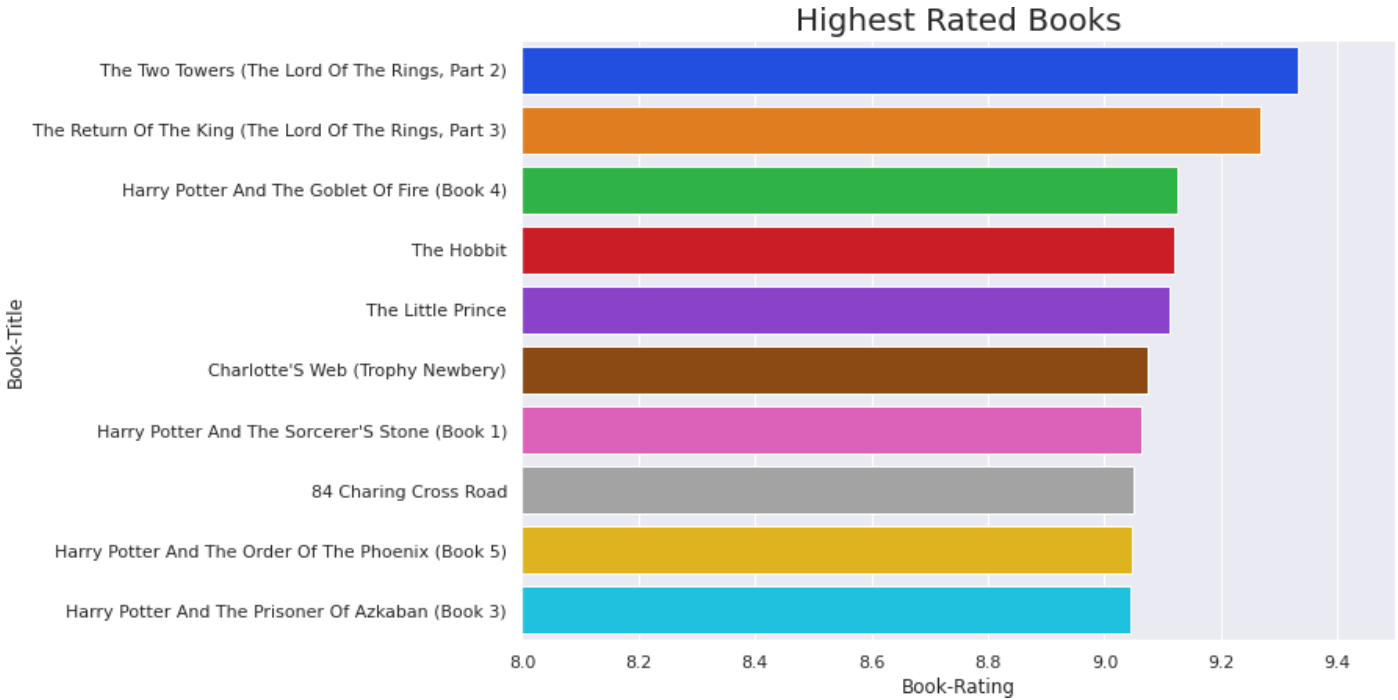
Here we can notice that the average rating given by the users is almost around 7 , also a surprisingly high number of books were awarded with a perfect score of 10 by the users

Most Popular Books



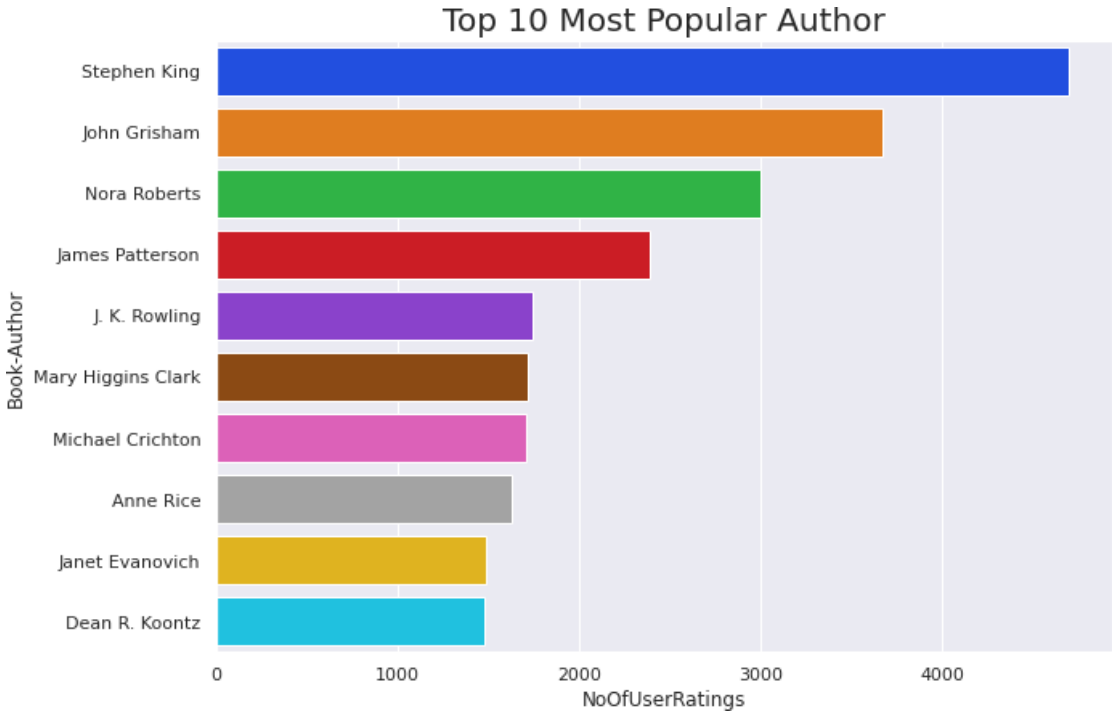
The Lovely Bones: A Novel is the most popular book in our dataset with more than 700 user ratings . Also just going by the name we can deduce that 2 books out of top 5 are novel which kind of points to the type of books more popular among users.

Highest Rated Books



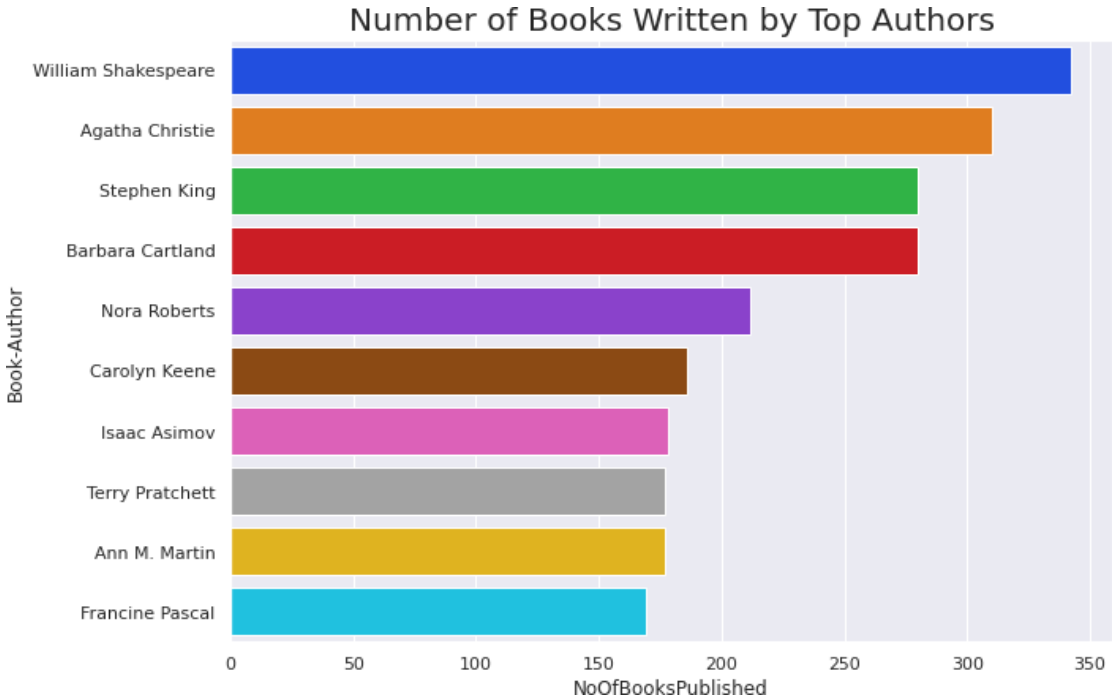
The Lord of the rings : The Two Towers is the highest rated book in our dataset . One thing worth noticing is that Harry Potter Series and Lord of the Ring series books are extremely loved by the users as they both in total account for 6 places out of this top 10 list of highest rated books

Most Popular Author



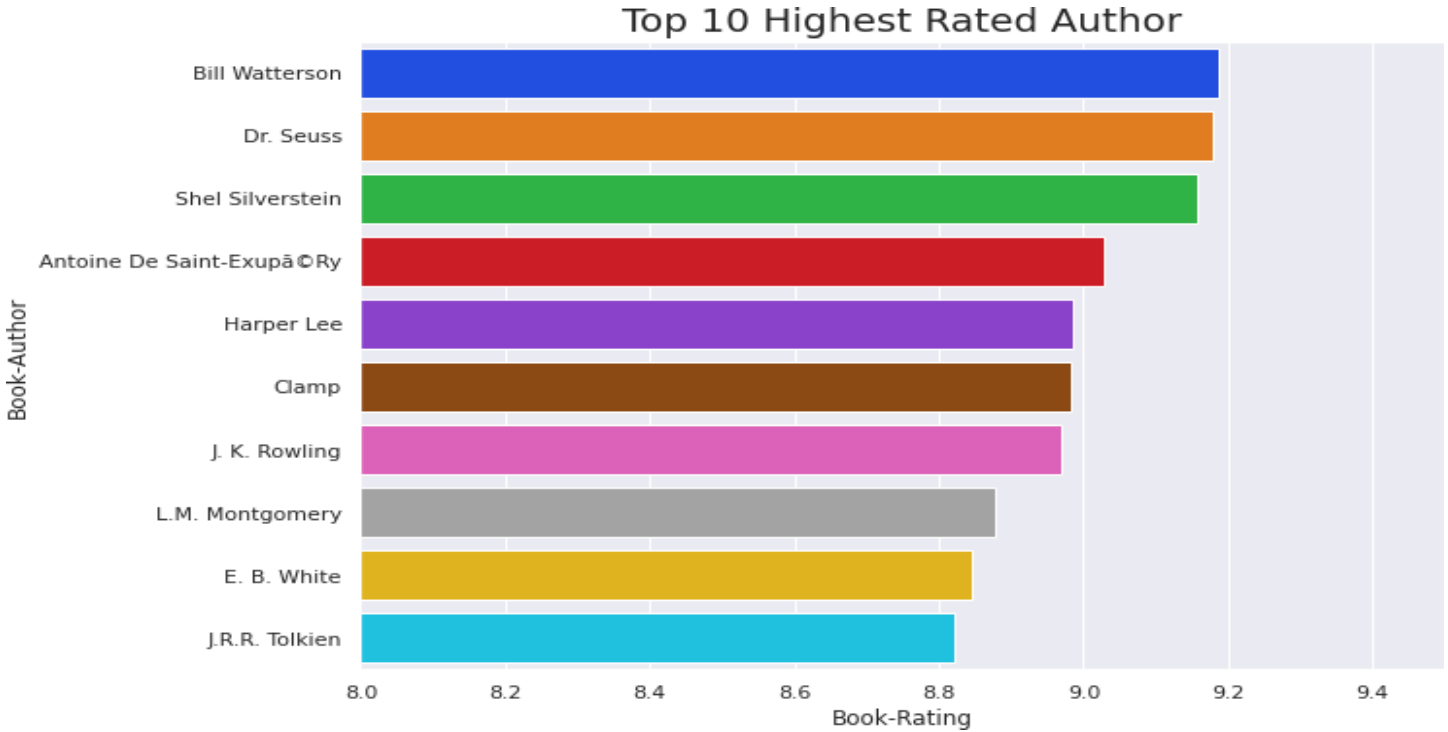
Stephen King is the most popular author and that too by a pretty healthy margin as per the data in our dataset . The combined user rating count for all of his books is more than 4600 which is at least 900 more than second placed John Grisham .

Most Books Written



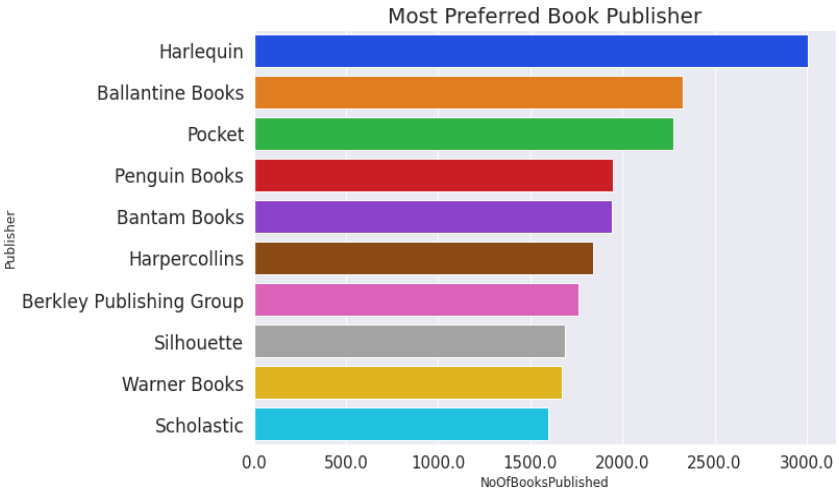
We have data for over 2.7 lakh books in our dataset out of which almost 330 books have been written by Sir William Shakespeare the most by any other author present in our dataset . On the second place is Agatha Christie and on third place is Stephen King who is also the most popular author as per then info present in our dataset

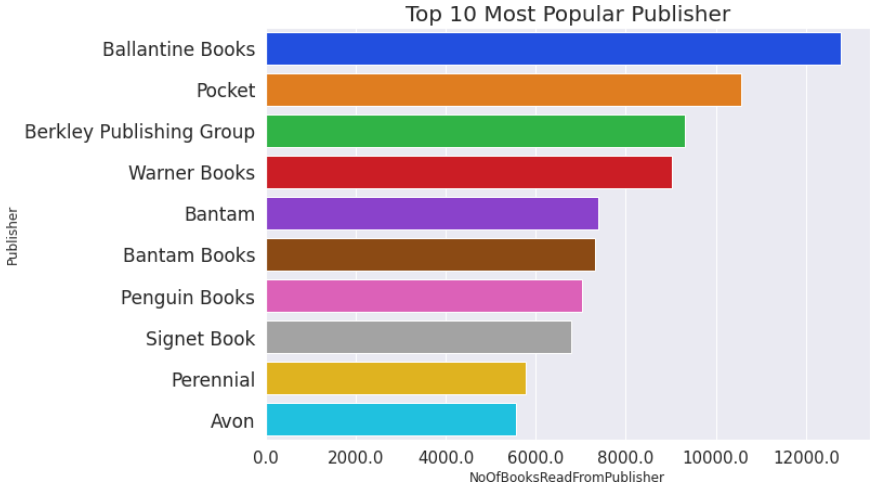
Highest Rated Author



Bill Watterson is the highest rated author in our dataset with his book receiving an average rating of around 8.7-8.9 . J.K Rowling and J.R.R Tolkien makes obvious entry into the list as former is the author of Harry Potter Series and later is the author of Lord Of The Rings series

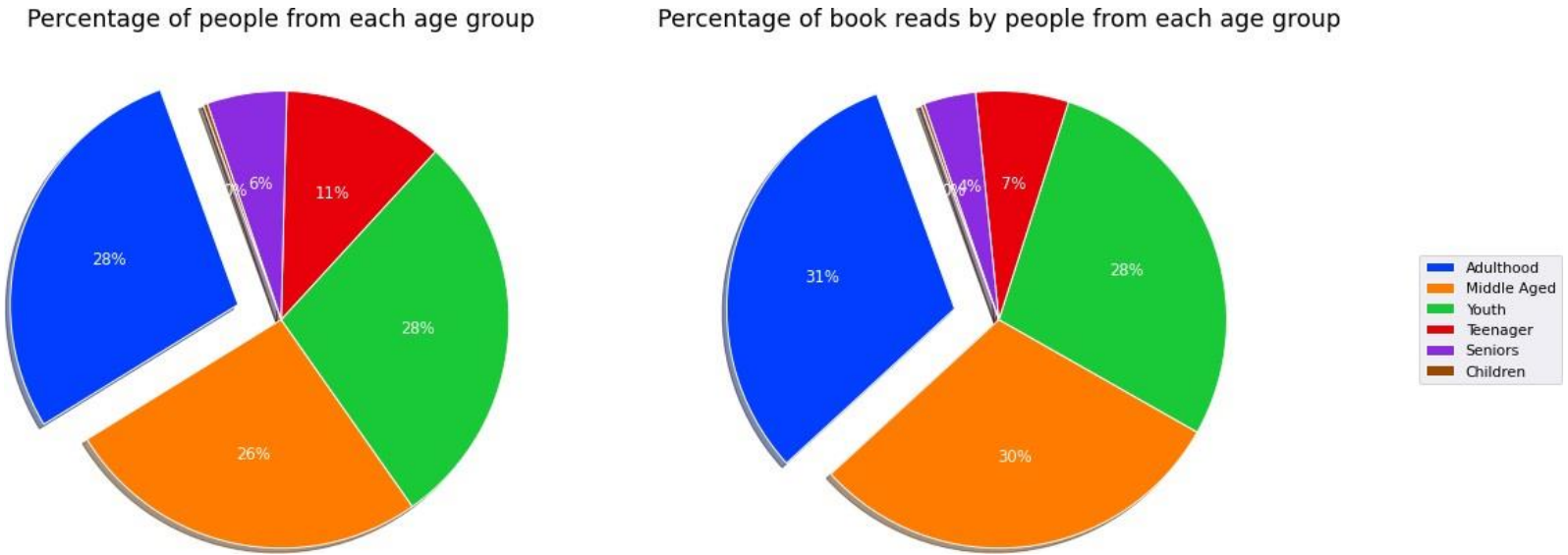
Book Publisher Analysis





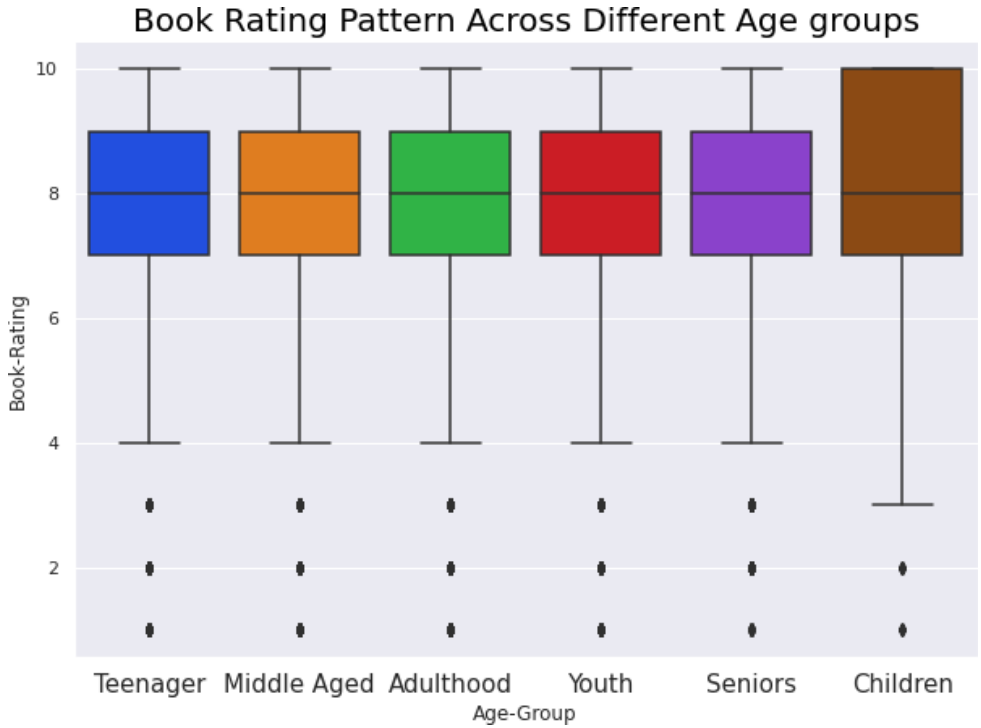
Our first bar plot is showing the list of publisher who have published the most number of books in our dataset and the top 3 are Harlequin, Ballantine Books and Pocket publishing house . The second bar plot is indicating the cumulative sum of book reads across each book published by a specific publisher and here Ballantine Books is on top which makes it the most popular book publisher .

Users Age Analysis



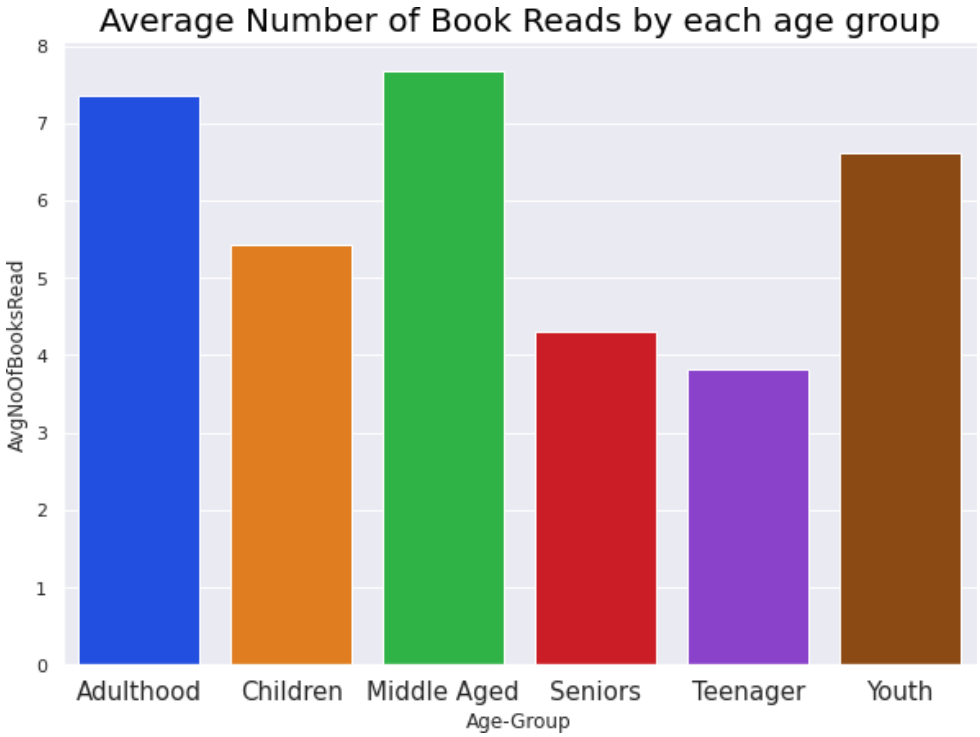
Here with the help of two pie plot we have tried to visualize the age distribution via age groups and their reading patterns . More than 80% of the users in our dataset belong to Youth , Middle Aged or Adulthood age group which basically covers 21-60 age brackets. People coming under Adulthood have read the most number of books , also one interesting thing to notice that even though 11 percent of the users are Teenager their books read share is just 7% .

Rating Pattern Across Age Groups



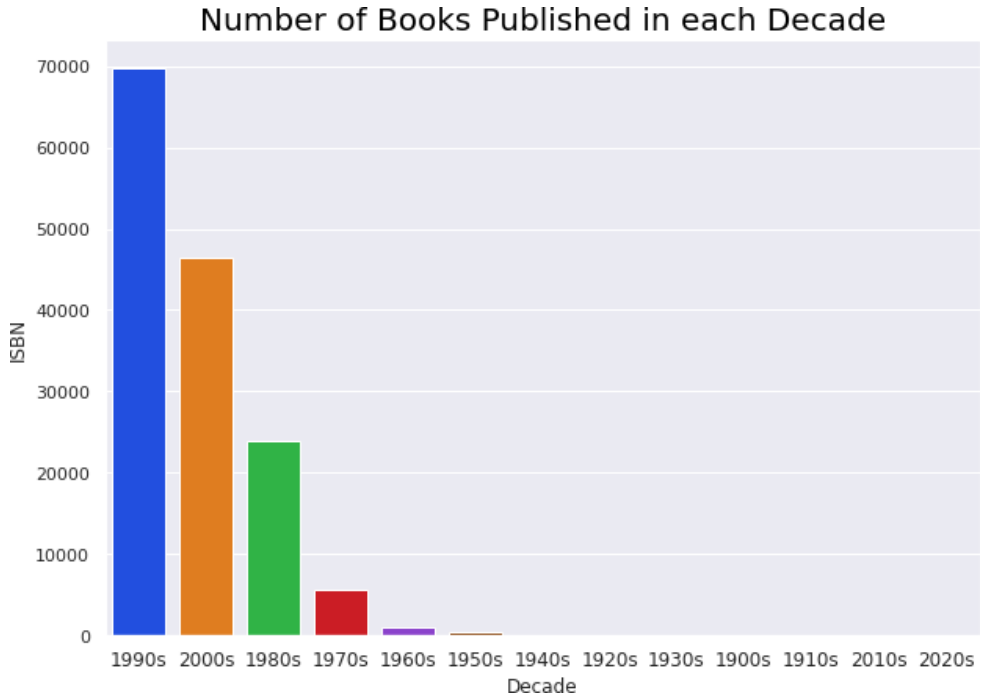
This box and whisker plot proves that there ain't much difference in rating pattern across people belonging from different age groups . The only take away from this box plot is that children are a bit more sentimental when it comes to rating books as compared to their elders.

Average Book Reads across Age groups



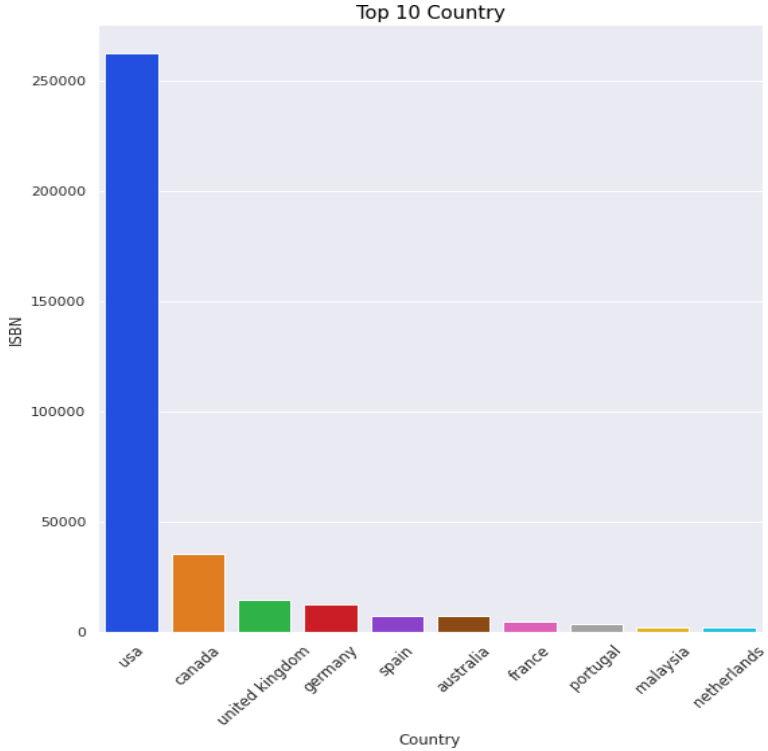
Using this bar plot we have tried to understand the average number of books read by a people from a specific age group . Here middle aged people are on top with an average of almost 8 book reads and on the bottom are Teenagers with an average book reads count of just below 4

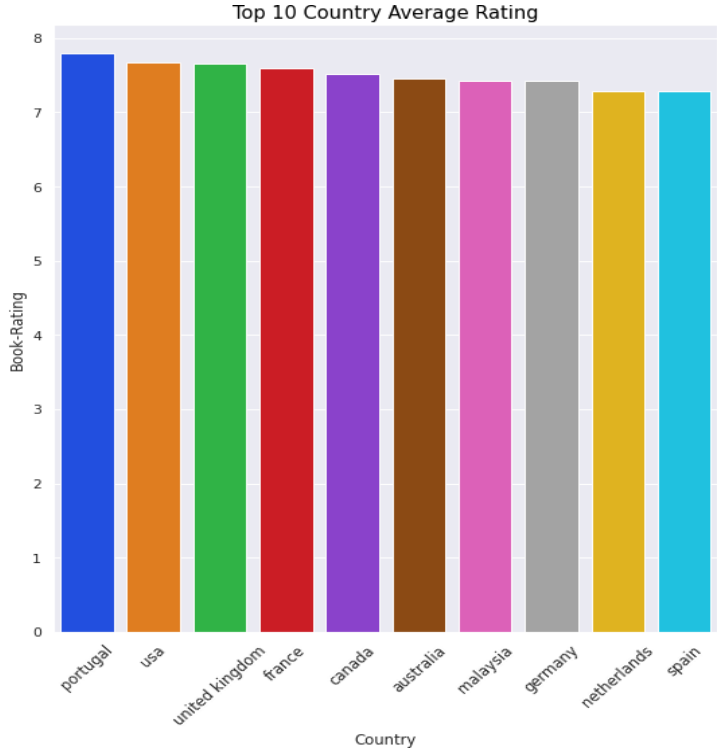
Year of Publication Analysis



Majority of the books in our dataset were published in the 90’s and 2000’s .
So our dataset doesn’t have much data of books published in recent years

Users Location Analysis





Majority of the users in our dataset are from USA and amongst this list of top 10 countries , users from Portugal are most lenient when it comes to rating the books they have read.

Popularity Based Recommendations

------ Top 10 Popular Recommendation for user id : 264543 are ------

	index	Book-Title	User-ID	Book-Rating
0	2210	The Lovely Bones: A Novel	187	8.240642
1	1981	The Da Vinci Code	144	8.569444
2	874	Harry Potter And The Chamber Of Secrets (Book 2)	121	8.826446
3	349	Bridget Jones'S Diary	116	7.534483
4	878	Harry Potter And The Prisoner Of Azkaban (Book 3)	113	9.132743
5	2333	The Red Tent (Bestselling Backlist)	109	8.532110
6	2255	The Nanny Diaries: A Novel	100	7.390000
7	876	Harry Potter And The Goblet Of Fire (Book 4)	100	9.190000
8	71	A Painted House	99	7.767677
9	175	Angels &Amp; Demons	99	8.070707

Our first recommender system is a rather simpler one which recommends the top 10 most popular unread books to a specific user . These are the Top 10 Popular Recommendation for user id 264543

Author Preference Based Recommendations

- Top 10 Popular Author Based Recommendation for user id : 136326 are -

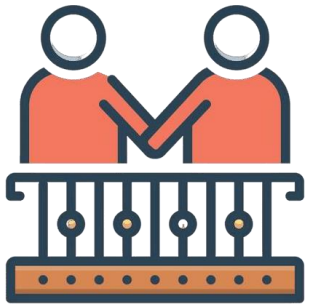
	index	Book-Title	User-ID	Book-Rating
0	43	Interview With The Vampire	83	7.698795
1	112	The Vampire Lestat (Vampire Chronicles, Book Ii)	67	8.268657
2	21	Dreamcatcher	60	7.616667
3	50	Misery	60	8.200000
4	98	The No. 1 Ladies' Detective Agency (Today Show...	52	8.461538
5	107	The Talisman	52	8.442308
6	41	Insomnia	51	8.274510
7	105	The Tale Of The Body Thief (Vampire Chronicles...	49	7.102041
8	108	The Tommyknockers	48	7.062500
9	81	The Dark Half	47	7.659574

For our second recommender system we are recommending the most popular unread piece of works from those author whose books the user has already read and liked in the past . These are the Top 10 Popular Author Based Recommendation for user id 136326

Recommendations Using Knn

```
['One For The Money (Stephanie Plum Novels (Paperback))',
 'The Pelican Brief',
 'Isle Of Dogs',
 'Hard Eight : A Stephanie Plum Novel (A Stephanie Plum Novel)',
 'Airframe',
 'A Place Called Freedom',
 'Cry Wolf',
 'The Associate',
 'Hot Six : A Stephanie Plum Novel (A Stephanie Plum Novel)',
 'Three To Get Deadly : A Stephanie Plum Novel (A Stephanie Plum Novel)',
 'Visions Of Sugar Plums: A Stephanie Plum Holiday Novel',
 'Wild Justice',
 'Bourne Identity',
 'Cause Of Death',
 "Mcnally'S Secret (Archy Mcnally Novels (Paperback))"]
```

```
--- Top 10 Recommendation for user 30035 are ---
1: High Five (A Stephanie Plum Novel) : 0.9571
2: Two For The Dough : 0.6844
3: Four To Score (A Stephanie Plum Novel) : 0.5935
4: Angus, Thongs And Full-Frontal Snogging: Confessions Of Georgia Nicolson : 0.5
5: Seven Up (A Stephanie Plum Novel) : 0.4963
6: Hard Eight : A Stephanie Plum Novel (A Stephanie Plum Novel) : 0.4625
7: My Antonia : 0.4507
8: Skipping Christmas : 0.3442
9: The Survivors Club : 0.3141
10: The Five People You Meet In Heaven : 0.3012
```



Here we are trying the find k users with the most similar to the user we are generating recommendation for and then we recommend books which has been already read and liked by these similar taste users . If we take user 30035 as an example then on left hand side we have plotted a list of books this user has already read and we can easily notice that this user likes to read books from **Stephanie Plum Novel** book series and our recommender system is rightfully recommending other books from the same series.

Matrix Factorization

The other two recommender system we are trying to build are based on a very popular collaborative filtering technique called Matrix Factorization . Our Recommender system has two entities : users and items(books) . If we have ***m*** users and ***n*** items(books) then the goal of our recommendation system was to build a ***m x n*** matrix (called the utility matrix) which consists of the rating or preference for each user-item pair . Initially, this matrix is usually very sparse because we only had ratings for a limited number of user-item pairs.

The matrix was populated by decomposing (or factorizing) the Utility matrix into two tall and skinny matrices. The decomposition has the equation:

$$\hat{Y} = UV^T$$

Where U is a ***m x k*** and V is ***n x k***. U is a representation of users in some low dimensional space, and V is a representation of items. For any user ***i*** and item ***j*** the rating prediction is simply .

$$\hat{y}_{ij} = u_i \cdot v_j$$

Evaluation Metrics

To evaluate our recommender system we randomly picked one already read and rated book for each of the user in our dataset and tossed that record into the test set whereas the train set contained all of the rest rating record for each of the user . Following are the few metrics we used to evaluate our recommender system

- **Hit Rate** : How often we are able to recommend a left-out rating
- **Cumulative Hit Rate** : Hit rate, confined to ratings above a certain threshold
- **Average Reciprocal Hit Rank** : Hit rate that takes the ranking into account.
- **Diversity** : 1 – S , where S is the average similarity score between every possible pair of recommendation for a given user .
- **Novelty** : Average popularity rank of recommended items.

Evaluating Recommendations

To Implement Matrix Factorization we used two methods to get the optimal decomposition .
Those two method are :

- SVD (Singular Value Decomposition)
- Gradient Descent

We evaluated the recommendations generated from both of these methods and the results are as follows :

	Algorithm	HR	cHR	ARHR	Diversity	Novelty
0	Gradient Descent	0.009479	0.009643	0.003576	0.961263	1488.934281
1	SVD	0.076619	0.080039	0.034478	0.860657	1294.716588

1. Hit Rate for SVD is around 7.6% whereas for Gradient Descent is just around 1%
2. Cumulative Hit Rate values for both of the methods are higher than hit rate which mean both of them are doing a slight better job at recommending the items users will actually like
3. Both Diversity and Novelty values for Gradient Descent method are higher than that of SVD which means recommendations generated from Gradient Descent method are more random in nature

Conclusions

- Majority of the book were rated above 7 by the users in our dataset
- Among the top 5 most read books , 2 of them were novels .
- Books from The Lord of the Rings and Harry Potter series are some of the most highly rated books in our dataset
- Among authors , Stephen King is most popular whereas if we talk about the number of books written then Sir William Shakespeare is on top.
- Most of the users in our dataset are from between the 21-60 age brackets.
- Books published by Ballantine books have been rated the most number of times in our dataset so they are like the most popular publisher , also in terms of total number of books published they are on second place.
- Also going by the data we can say that users who belong to 30-50 age brackets are more into reading books as compared to teenagers
- Most of the book reading users are from USA followed by Canada on second place .
- Going by the offline evaluation , recommendations generated via SVD were the most interesting with a good mix of popular and novel books and a good Hit Rate of 7.5 % as well.

Challenges Faced

- All the three datasets provided to us were really huge with number of records ranging above the 2 lakh mark for each of them
- Most of the book rating information in our ratings dataset were implicit
- The user item matrix that was generated to apply Matrix Factorization on , was highly sparse
- Evaluating the Recommendations offline was the most biggest challenge as it was extremely hard to find meaningful offline evaluation metrics .