Capstone Project
Book Recommendation System

By - Dishant Toraskar

Content

Problem Statement

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 (items being movies to watch, text to read, products to buy, or anything else depending on industries).
Recommender systems are really critical in some industries as they can generate a huge amount of income when they are efficient or also be a way to stand out significantly from competitors. The main objective is to create a book recommendation system for users.

Data Description

The Book-Crossing dataset comprises 3 files -

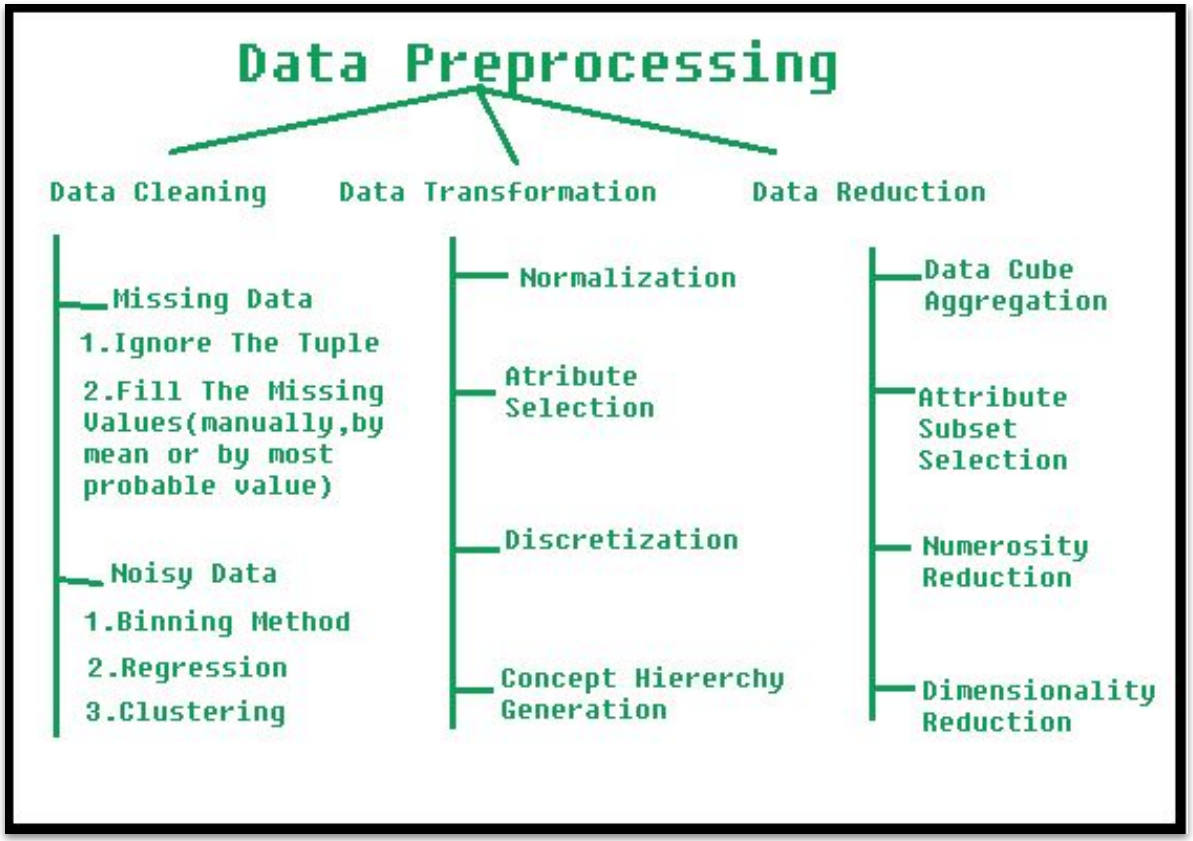
- Users : Contains the users. Note that user IDs (User-ID) have been anonymized and map to integers. Demographic data is provided (Location, Age) if available. Otherwise, these fields contain NULL values.

- Books : Books are identified by their respective ISBN. Invalid ISBNs have already been removed from the dataset. Moreover, some content-based information is given (Book-Title, Book-Author, Year-Of-Publication, Publisher), obtained from Amazon Web Services. Note that in the case of several authors, only the first is provided. URLs linking to cover images are also given, appearing in three different flavors (Image-URL-S, Image-URL-M, Image-URL-L), i.e., small, medium, large. These URLs point to the Amazon website.

- Ratings : Contains the book rating information. Ratings (Book-Rating) are either explicit, expressed on a scale from 1-10 (higher values denoting higher appreciation), or implicit, expressed by 0.

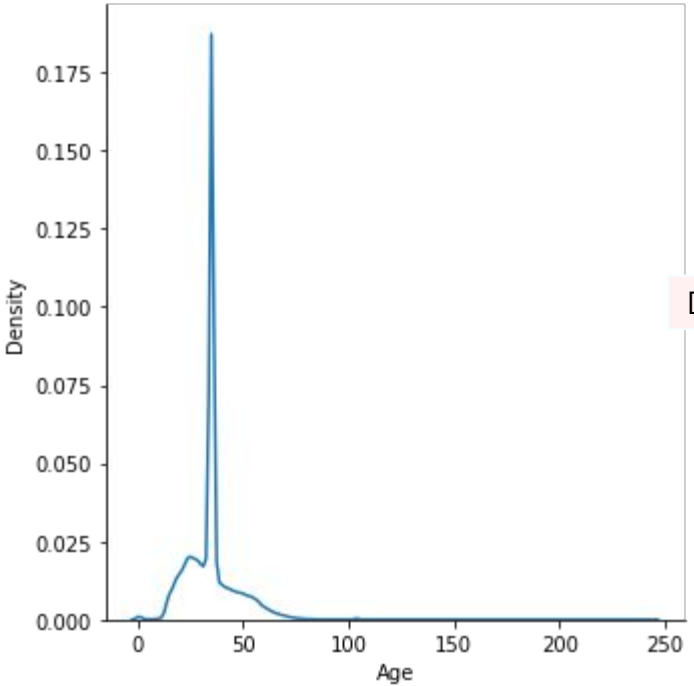
Data Preprocessing

Data preprocessing is a technique which is used to transform raw data into clean, efficient and useful data.

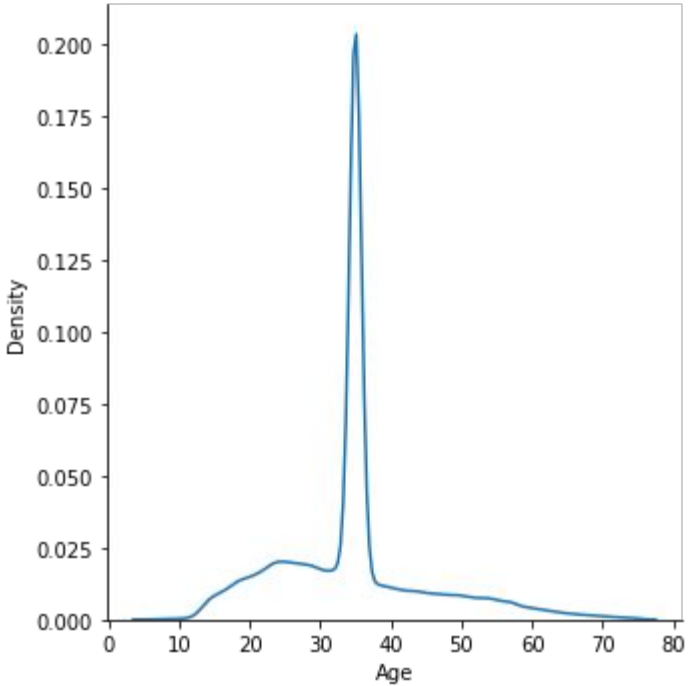


Exploratory Data Analysis

Age

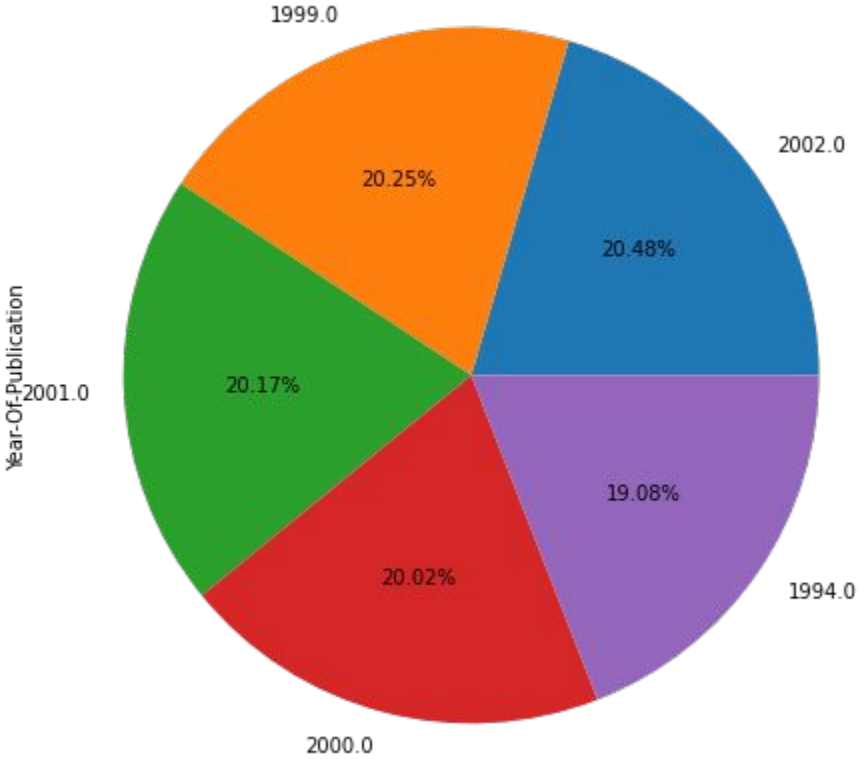


Data Preprocessing



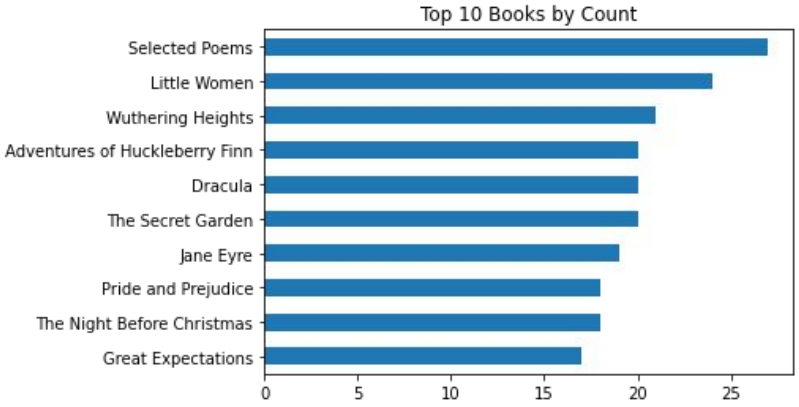
Most number of book readers are between the age group 30 - 40.

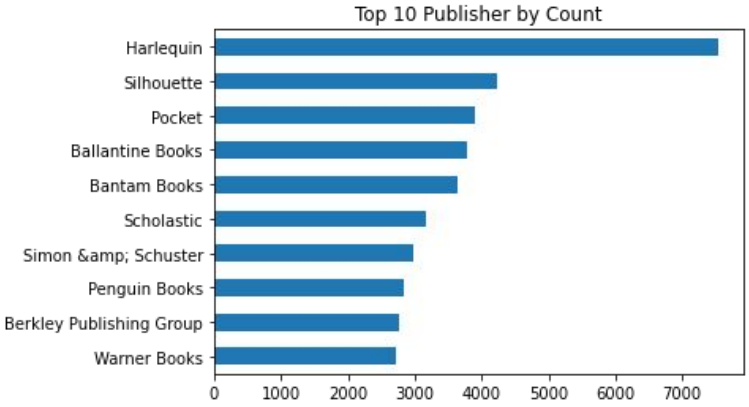
Books - Year of Publishing

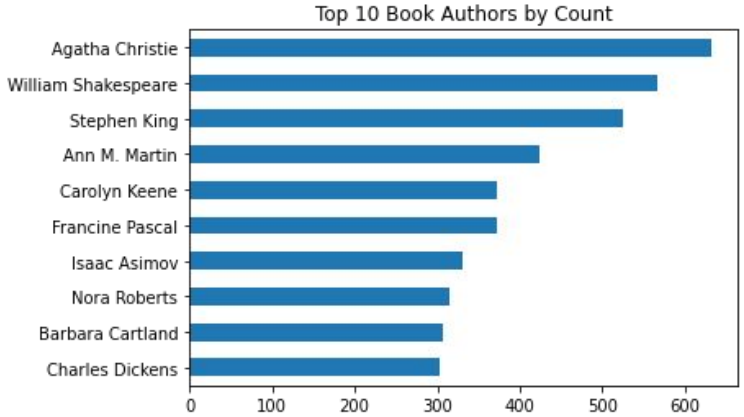


Most number of books are published in the year 2002, followed by 1999 & 2001.

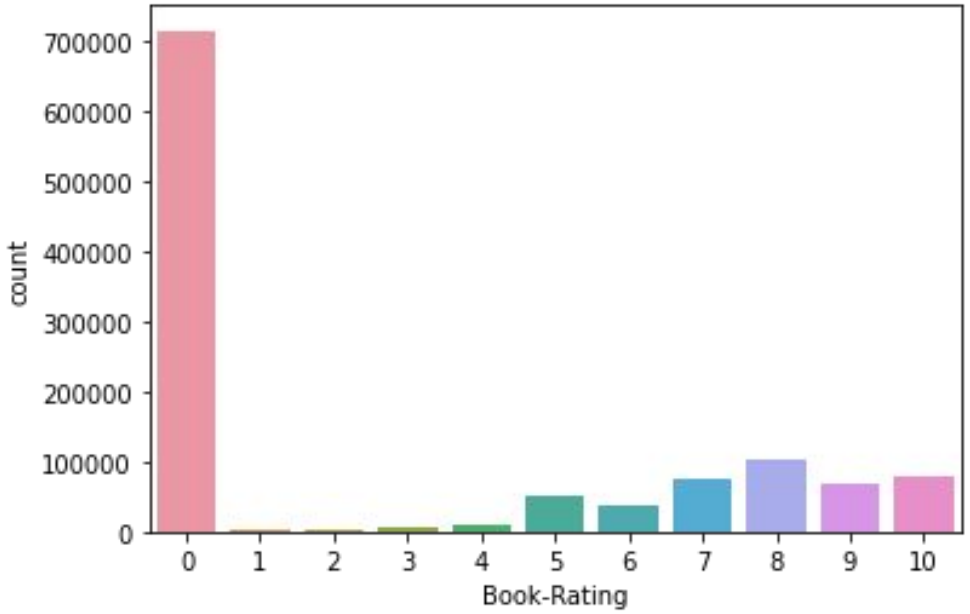
Top 10 Books, Publisher & Author





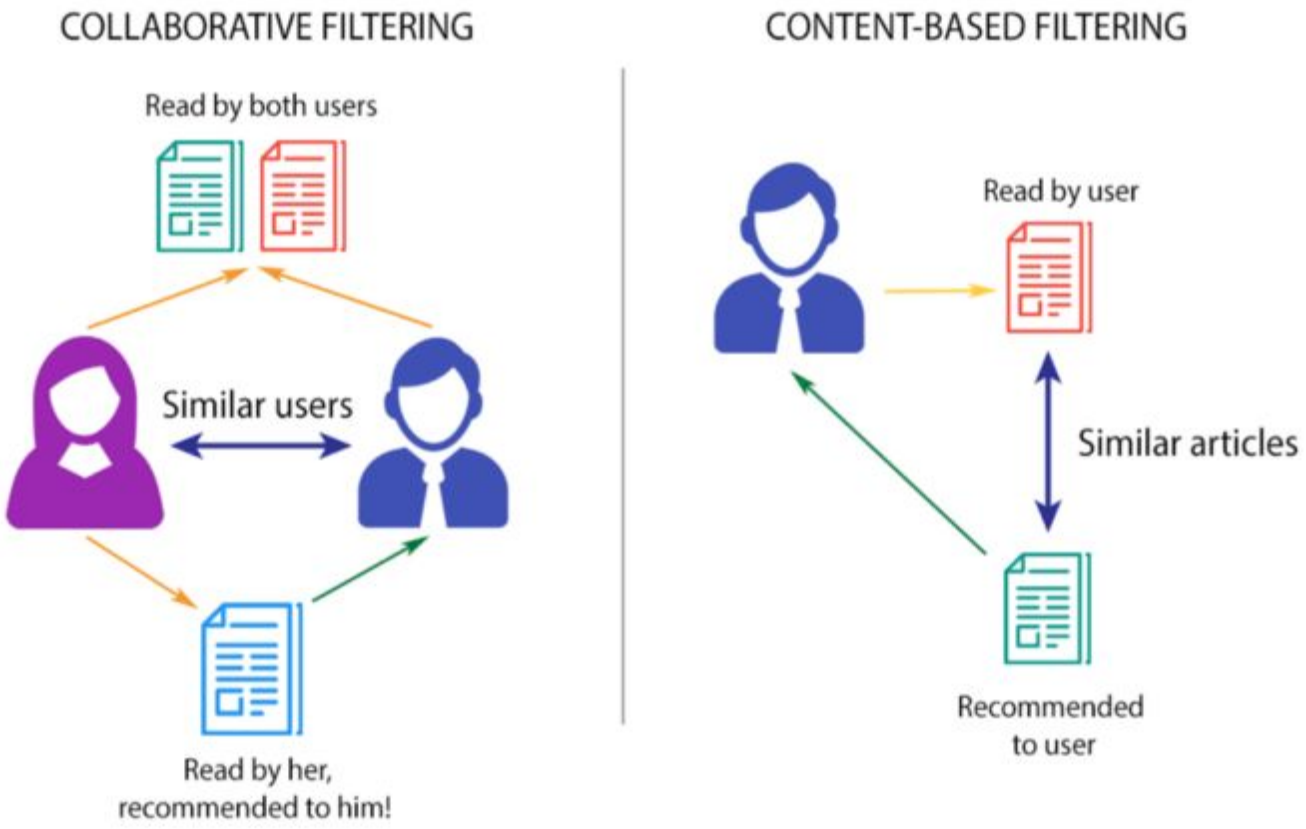


Number of Ratings



We can observe that most of the book are not rated. Thus rating is 0. We will ignore such books while building or training a recommendation system. Beside this, 8 is the highest number of ratings count received to book followed by 7, 10 & 9.

Types of Recommendation System



Book Recommendation System Using k-NN

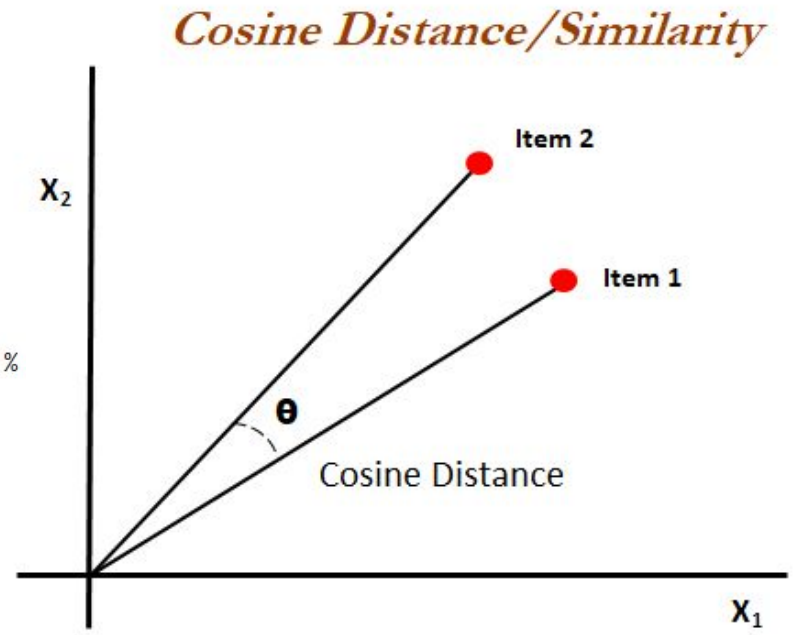
Cosine Similarity

We will use cosine similarity with respect to kNN. kNN helps us to compute the distance between two points (consider books as a point here). So, we can use cosine to find the angle of line between two distance. Cosine similarity gives us the similarity score between two points(books in this case).

Example -

```
Recommendation for 1984
1) Animal Farm , with similarity of 23.4 %
2) American Psycho (Vintage Contemporaries) , with similarity of 19.38 %
3) The Hitchhiker's Guide to the Galaxy , with similarity of 18.66 %
4) Brave New World , with similarity of 18.18 %
```

We will get the distance(0-1) between two points but for ease of understanding I have subtracted distance with 1 to get how much similar two books are.

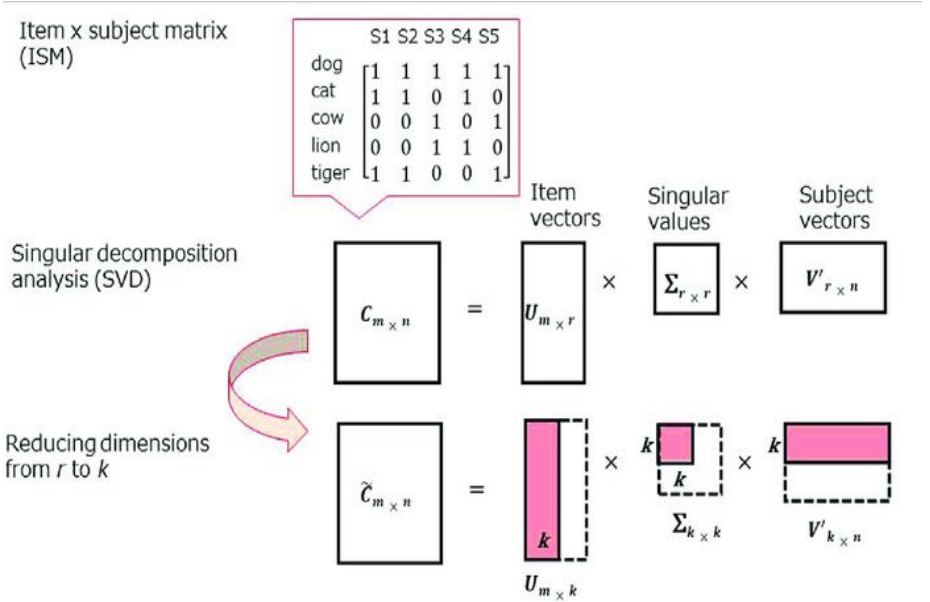


Model Based Collaborative Filtering

SVD - Latent Factor Model Collaborative Filtering

SVD or Singular Value Decomposition a method from linear algebra that has been generally used as a dimensionality reduction technique in machine learning. SVD is a matrix factorisation technique, which reduces the number of features of a dataset by reducing the space dimension from N-dimension to K-dimension (where K < N).

In the context of the **recommendation system**, the SVD is used as a collaborative filtering technique. It uses a matrix structure where each row represents a user(User-ID in this case), and each column represents an item(Book ISBN in this case). The elements of this matrix are the ratings that are given to items by users(Book rating given by user). It provides another way to factorize a matrix, into singular vectors and singular values.



Evaluation Using Top - N Metrics

Recommender Model Evaluation

In Recommender Systems, there are a set metrics commonly used for evaluation. We choose to work with **Top-N accuracy metrics**, which evaluates the accuracy of the top recommendations provided to a user, comparing to the items the user has actually interacted in test set.

This evaluation method works as follows:

- For each user
  - For each item the user has interacted in test set
    - Sample 100 other items the user has never interacted.
    - Ask the recommender model to produce a ranked list of recommended items, from a set composed of one interacted item and the 100 non-interacted items
    - Compute the Top-N accuracy metrics for this user and interacted item from the recommendations ranked list
- Aggregate the global Top-N accuracy metrics

How Recommendation System Performed?

Global metrics:
{'modelName': 'Collaborative Filtering', 'recall@5': 0.2566985645933014, 'recall@10': 0.35909090909090907, 'recall@15': 0.4368421052631579}

	hits@5_count	hits@10_count	hits@15_count	interacted_count	recall@5	recall@10	recall@15	User-ID
127	5	6	10	34	0.147059	0.176471	0.294118	16795
74	5	7	10	29	0.172414	0.241379	0.344828	95359
45	10	12	14	28	0.357143	0.428571	0.500000	104636
77	8	8	9	22	0.363636	0.363636	0.409091	153662
10	4	5	6	21	0.190476	0.238095	0.285714	158295
278	15	17	17	20	0.750000	0.850000	0.850000	114368
216	9	10	14	19	0.473684	0.526316	0.736842	258534
223	3	5	7	19	0.157895	0.263158	0.368421	60244
23	8	10	11	17	0.470588	0.588235	0.647059	140358
66	1	1	2	16	0.062500	0.062500	0.125000	135149

recall@5 - 25% of the items were interacted by the user in test set from top 5 recommendations.

recall@10 - 35% of items were interacted from top 10 recommendations.

recall@15 - 43% of items were interacted.

Future Scope

1. We can use content - based filtering on the available text columns such as book-title, publisher & author to come up with a content based recommendation system.
2. We can use NLP techniques such as TF-IDF for performing content - based filtering.
3. We can try to collect more data from various other sources if possible.
4. We can try to add description for the book so that content - based filtering can easily be performed.
5. We can also try to get reviews by the customer for every books.
6. Get more understanding about the domain and research more about recommendation system.

Conclusion

1. Started with EDA for the given 3 dataset of books, users & ratings.
2. Most number of books are read by people between 30 to 40 age.
3. 2002 have the highest number of books published followed by 1999.
4. Selected Poems is the book having most number of readers.
5. Harlequin is the publisher for approximately around 7500 books.
6. Agatha Christie is the Author having most number of book.
7. Highest rating is 8 out of 10. Which means most of the books are rated 8 by most of the users. 1 & 2 are on the lowest side (0 Rating is ignored).
8. Used k-NN to find similarity between users reading book using cosine similarity.
9. Trained model only for users from USA & Canada due to speed & time issues.
10. Selected users having 10 or more interactions.
11. Performed dimensionality reduction using SVD.
12. Built a model-based collaborative filtering recommendation class.
13. User Top-N metric for evaluation of recommender system.
14. Performance -
    - Recall@5 - 25%.
    - Recall@10 - 35%.
    - Recall@15 - 43%.

THANK YOU