Capstone Project - 4

Book Recommendation System

Team Members

Amir Khan
Saurabh Daund
Het Kothari
Kamya Malhotra
Mouleena Jaiswal

Table of contents:

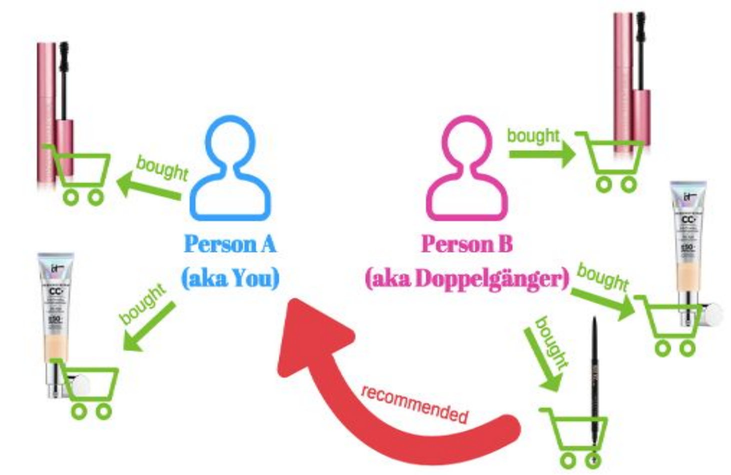
- Introduction
- Defining Problem Statement
- Data Overview
- Exploratory Data Analysis
- Model Creation
- Model Evaluation

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

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 Overview

- ➢ Understanding datasets better:

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.

Continued...

- ➢ Understanding attributes of data set better:
  - Books:'ISBN', 'Book-Title', 'Book-Author', 'Year-Of-Publication', 'Publisher', 'Image-URL-S', ‘Image-URL-M', 'Image-URL-L'
  - Users:’User-ID’,’Location’,’Age’
  - Ratings:’User-ID’,’ISBN’,’Book Rating’
- ➢ Summarize the data by identifying key characteristics, such as data volume and total number of variables in the data.

Book :**Number of rows in our dataset are 271360**

Number of columns in our dataset are 8

Users:**Number of rows in our dataset are 278858**

Number of columns in our dataset are 3

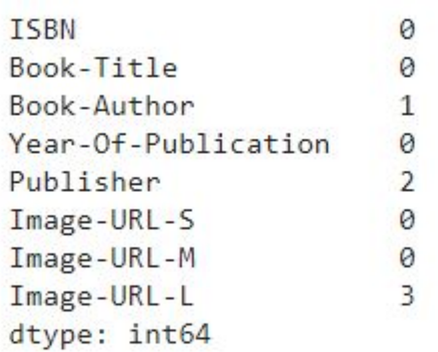
Ratings:**Number of rows in our dataset are 1149780**

Number of columns in our dataset are 3

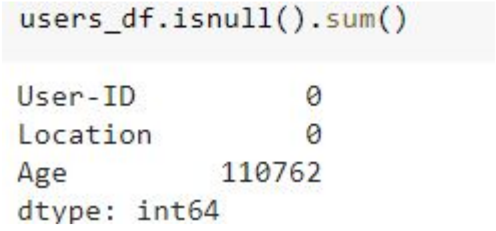
Continued...

- ➢ Understand the problems with the data, such as missing values, inaccuracies, and outliers.

Book Users Ratings

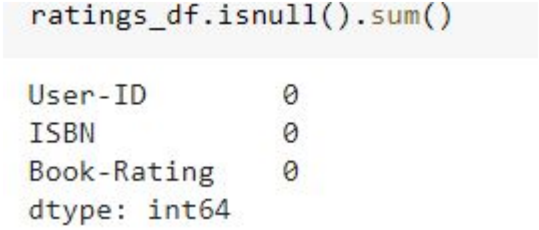


```
ISBN                   0
Book-Title             0
Book-Author            1
Year-Of-Publication    0
Publisher              2
Image-URL-S            0
Image-URL-M            0
Image-URL-L            3
dtype: int64
```



```
users_df.isnull().sum()
```

```
User-ID          0
Location         0
Age         110762
dtype: int64
```



```
ratings_df.isnull().sum()
```

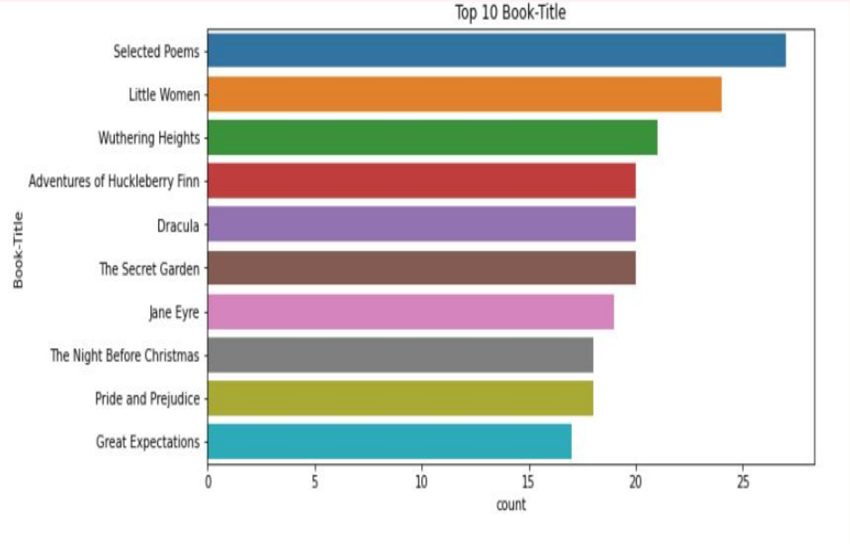
```
User-ID        0
ISBN           0
Book-Rating    0
dtype: int64
```

- ➢ There are NAN/NULL values in our book dataset and we have dropped these values.

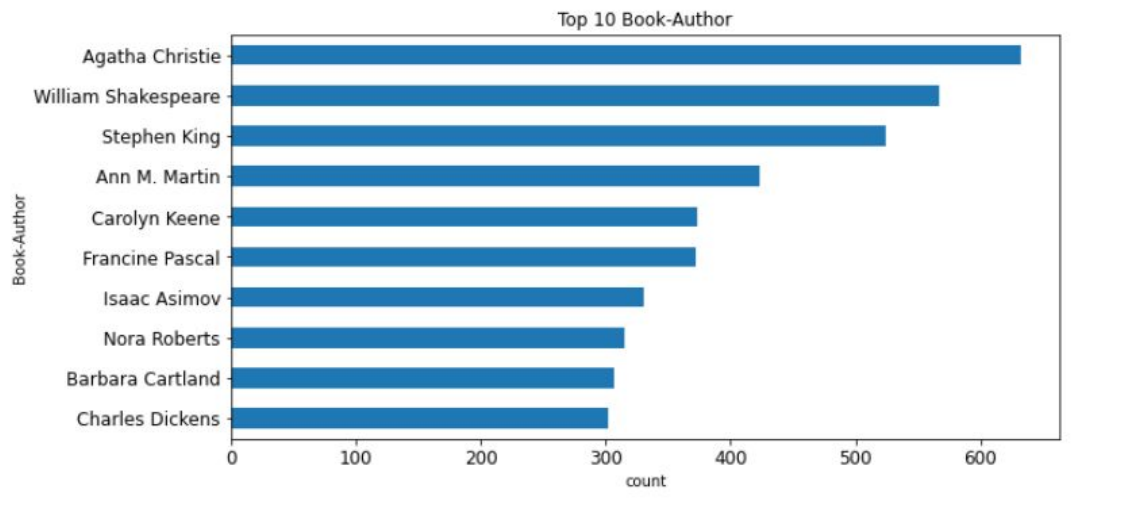
EDA

1. Book Title

- We have 2,42,130 unique Book Title.
- The top most book title found is Selected Poems.
- Selected Poems is present in 27 rows in books data.

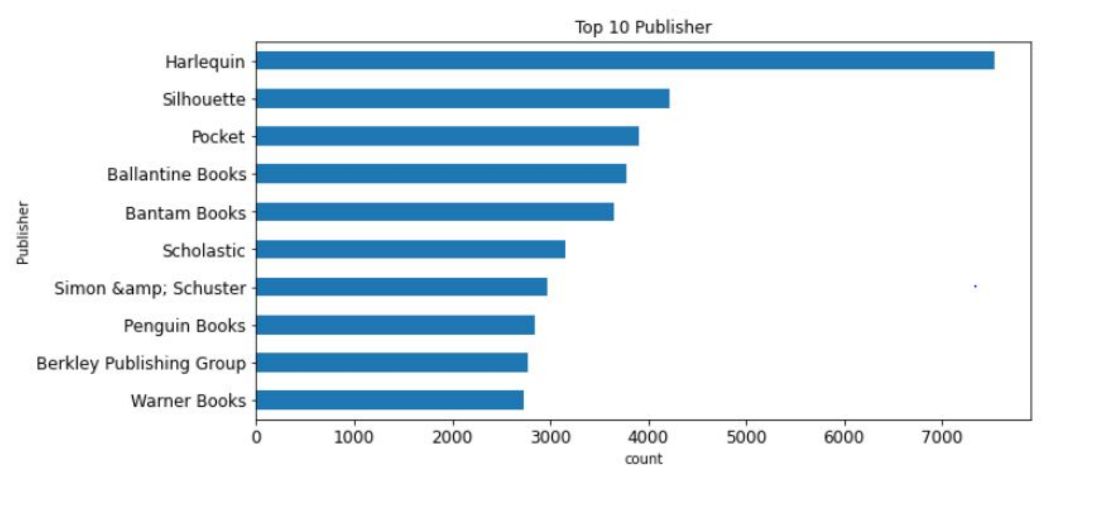


2. Book Author



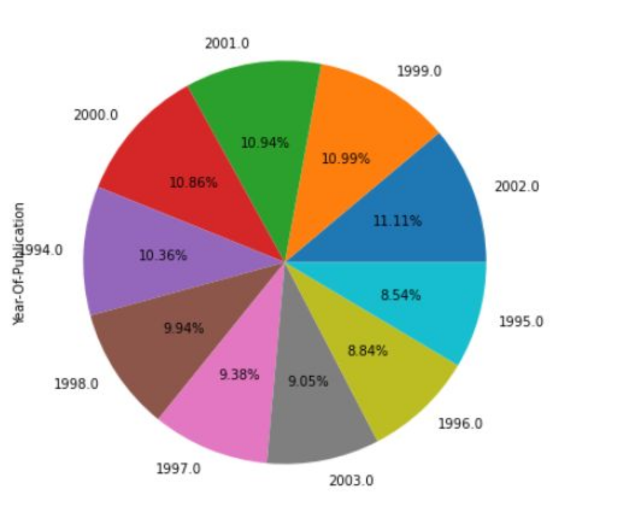
- We have 1,02,020 unique Author.
- The top most author found is Agatha Christie.
- We have 632 Agatha Christie in books data.

3. Publisher



- **We have 16,803 unique Publishers.**
- **The top most publisher found is Harlequin.**
- **We have 7,535 Harlequin in books data.**

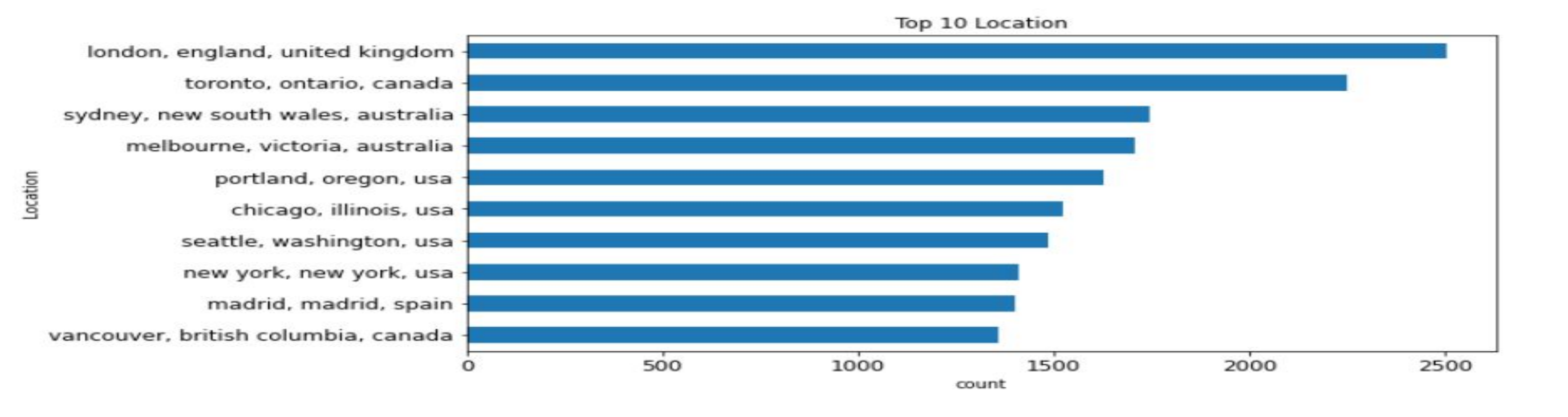
4. Year of Publication



Most books were published in the year of 2002.

EDA

5. Location

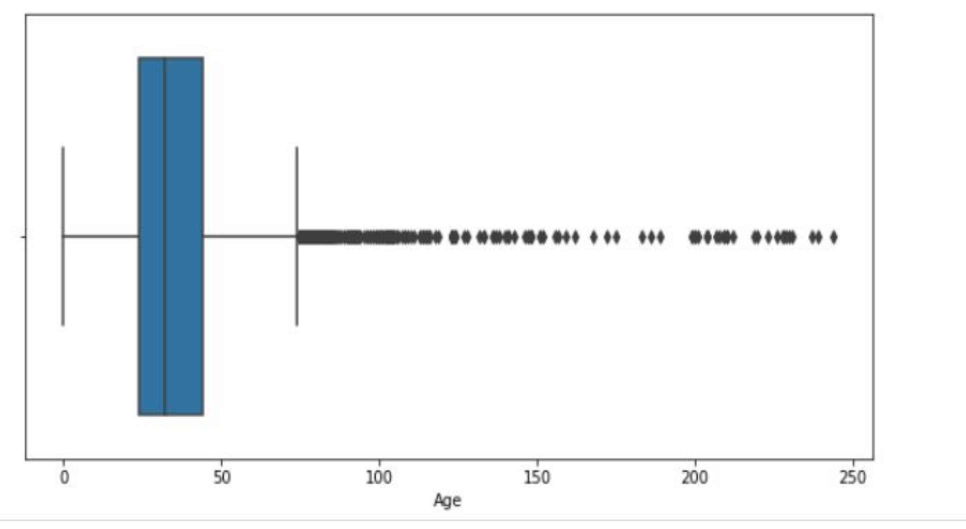


- In these plot , we can see top 10 users location.
- The top most location is London, England, United Kingdom
- We have around 2500 users from London.

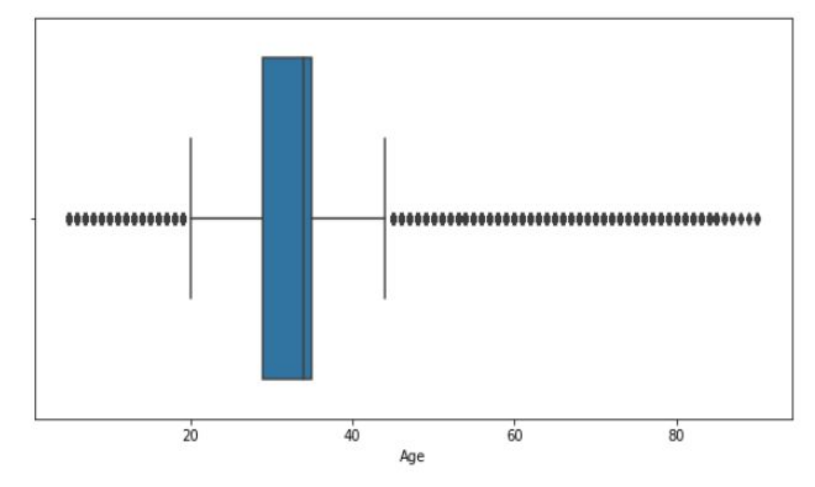
EDA

6. Age

Before Age Outlier Removal

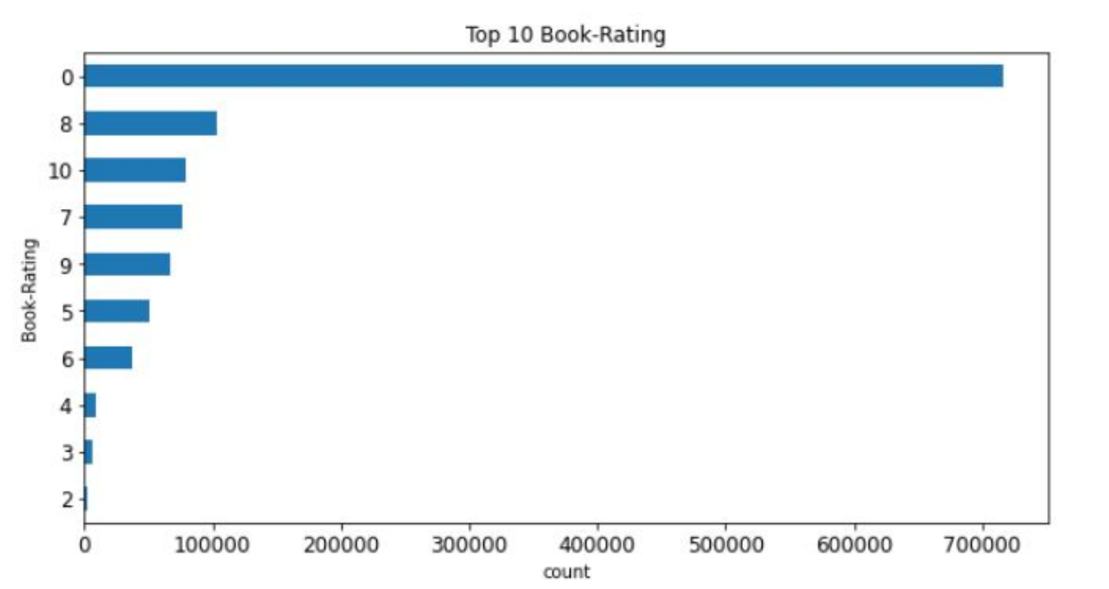


After Age Outlier Removal



- **Remove the Age greater than 90 and less than 5.**

EDA



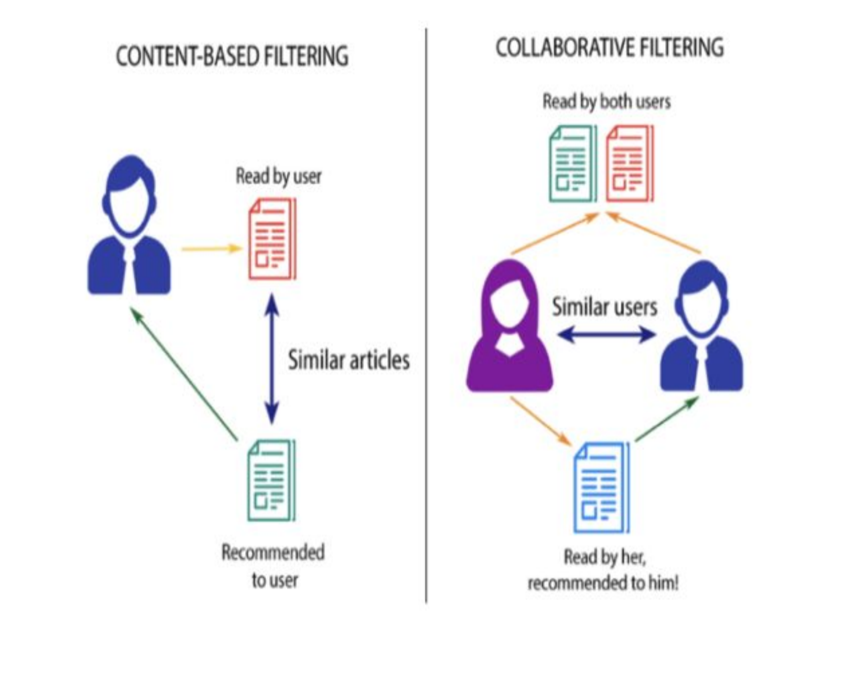
- Highest rating is 8 out of 10.
- Lowest rating is 2 out of 10.
- Most of the people haven't given the ratings.

.

Model Creation

Recommendation system is usually classified on rating estimation:

- ➢**Collaborative Filtering system**
- ➢**Content based system**
- ➢**Hybrid based system**



Collaborative Filtering Using k-Nearest Neighbors (kNN)

kNN 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-k nearest neighbors.

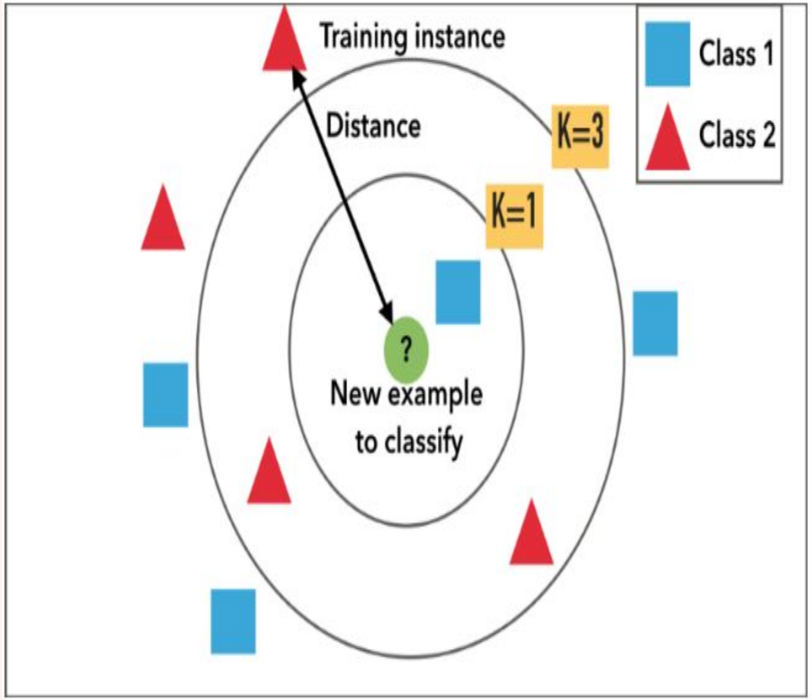
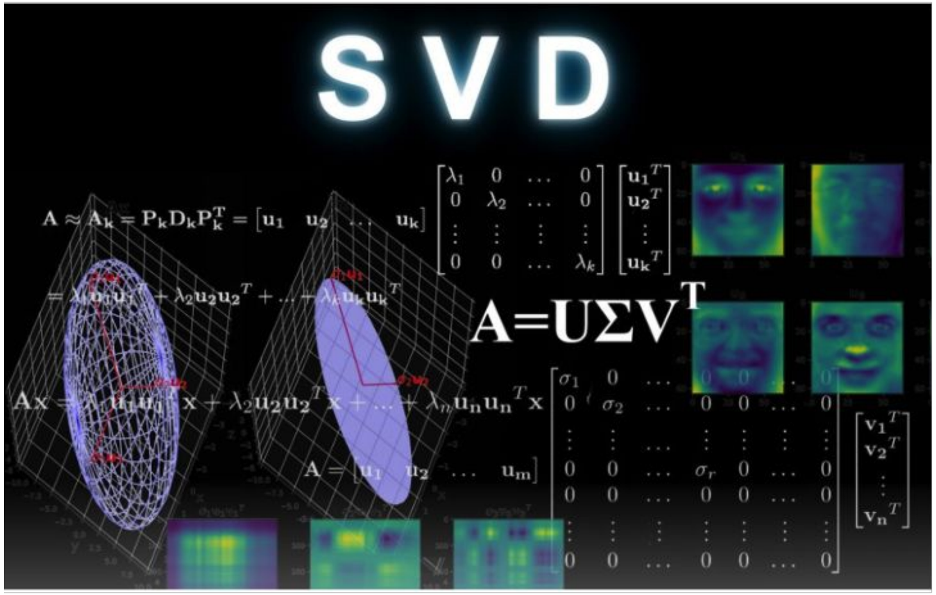


Illustration of how KNN makes classification about new sample

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.


S V D
$A \approx A_k = P_k D_k P_k^T = [u_1 \ u_2 \ \dots \ u_k]$
$= \lambda_1 u_1 u_1^T + \lambda_2 u_2 u_2^T + \dots + \lambda_k u_k u_k^T$
$A = U \Sigma V^T$
$Ax = \lambda_1 u_1 u_1^T x + \lambda_2 u_2 u_2^T x + \dots + \lambda_n u_n u_n^T x$
$A = [u_1 \ u_2 \ \dots \ u_m]$

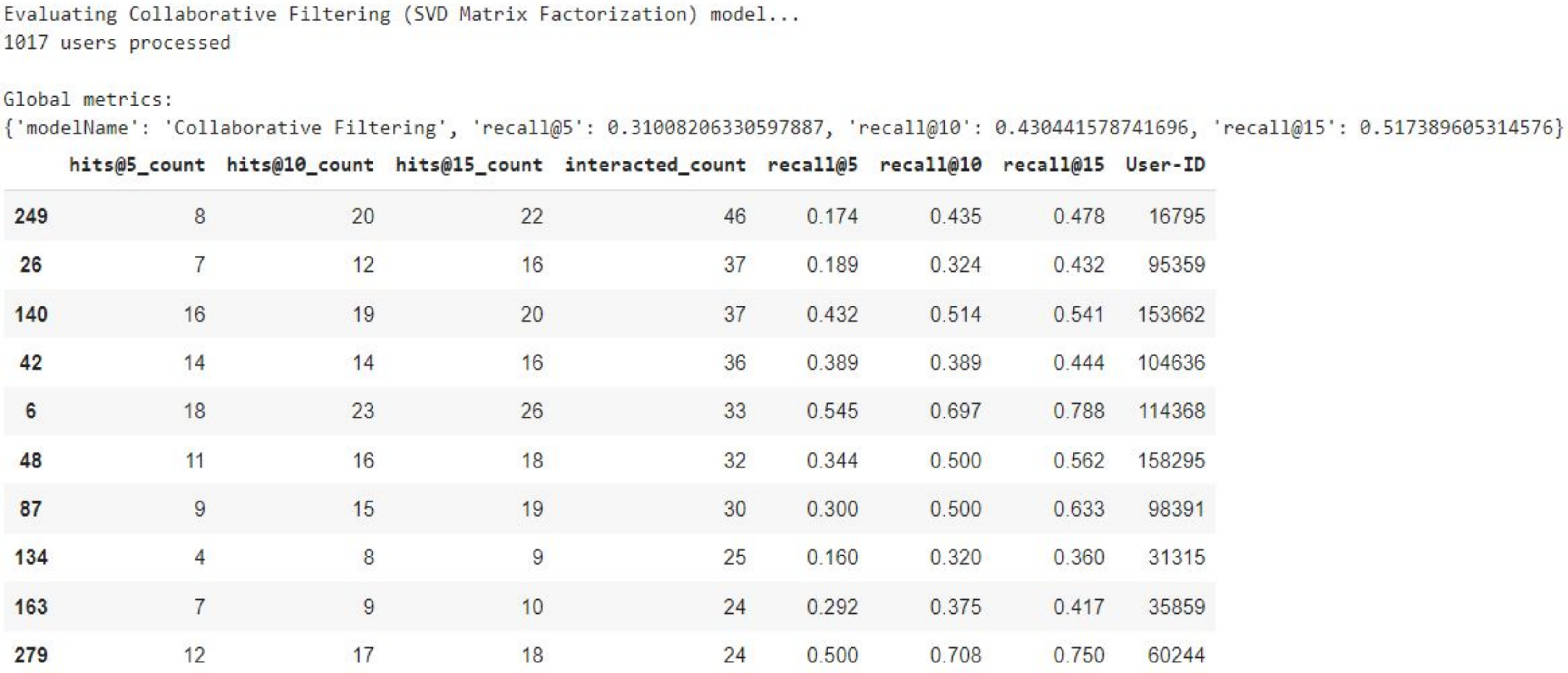
Evaluation

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

Evaluation Contd.



```
Evaluating Collaborative Filtering (SVD Matrix Factorization) model...
1017 users processed

Global metrics:
{'modelName': 'Collaborative Filtering', 'recall@5': 0.31008206330597887, 'recall@10': 0.430441578741696, 'recall@15': 0.517389605314576}
```

	hits@5_count	hits@10_count	hits@15_count	interacted_count	recall@5	recall@10	recall@15	User-ID
249	8	20	22	46	0.174	0.435	0.478	16795
26	7	12	16	37	0.189	0.324	0.432	95359
140	16	19	20	37	0.432	0.514	0.541	153662
42	14	14	16	36	0.389	0.389	0.444	104636
6	18	23	26	33	0.545	0.697	0.788	114368
48	11	16	18	32	0.344	0.500	0.562	158295
87	9	15	19	30	0.300	0.500	0.633	98391
134	4	8	9	25	0.160	0.320	0.360	31315
163	7	9	10	24	0.292	0.375	0.417	35859
279	12	17	18	24	0.500	0.708	0.750	60244

Conclusion

- As we can see, after implementing Collaborative Filtering and evaluating it using SVD matrix we are satisfied with the results. A recall rate of around 50 for hit@15 is fair enough for such a large dataset. Also since it is an unsupervised learning algorithm trying to find good books that users will like which in itself is a very vast and complicated study.

Challenges

➢High Volume of Data.

➢Elevating evaluation score for the models.

➢Crashing of session due to large pivot matrix.

➢Choosing optimal number of Factors in SVD.

THANK YOU!!