أكاديمية سدايا
SDAIA Academy

Book Recommendation System

Prepared by: Shaden Almangour

Outlines

01 Introduction

02 Approach

03 Result

04 Tools

05 Conclusion

Introduction

01

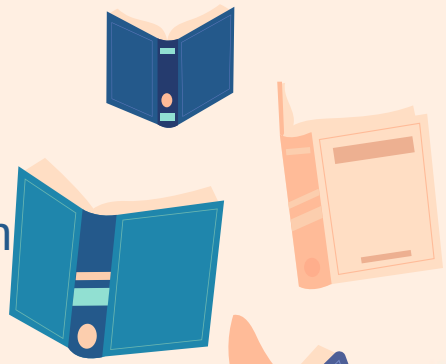
Introduction

The explosive growth in the amount of available information, and the number of visitors to the Internet has created a potential challenge of information overload which hinders timely access to items of interest. This has increased the demand for recommender systems more than ever before.

Introduction (Cont.)

Recommender systems are information filtering systems that deal with the problem of information overload by filtering vital information fragment out of a large amount of dynamically

generated information according to the user's preferences, interests, or observed behaviour about the item.



Problem statement

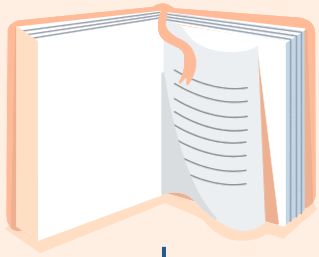
People may have difficulties in choosing the right books or searching for books based on their interests, and this may take a long time.

The solution

To deal with this problem, the solution is a recommendation system that helps the user by providing the search for relevant suggestions to the users. This process is done by filtering based on the ratings of other users.

About the recommendation system

Recommendations typically speed up searches and make it easier for users to choose the books their other users were reading.



Suggestions

Popular books might you like

Recommended for you!

Most of the users have read this book, you might want to read it

Related books

Similarity book for your last read might you like .

02

Approach

Work Approach

Step 1
Data Collection

Step 2
Data Clean

Step 3
EDA

Step 4
Modeling

Step 5
Recommendation system

Step 1: Data Collection

About the dataset:

The data is collected from the Kaggle which is contains from 3 csv files

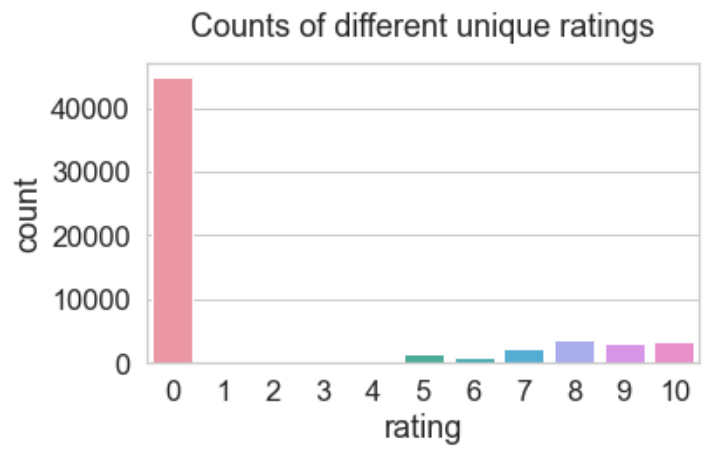
About the dataset:

- Books: information of each book.
- Ratings: User rating records for each book.
- Users: User information (Age, Location)

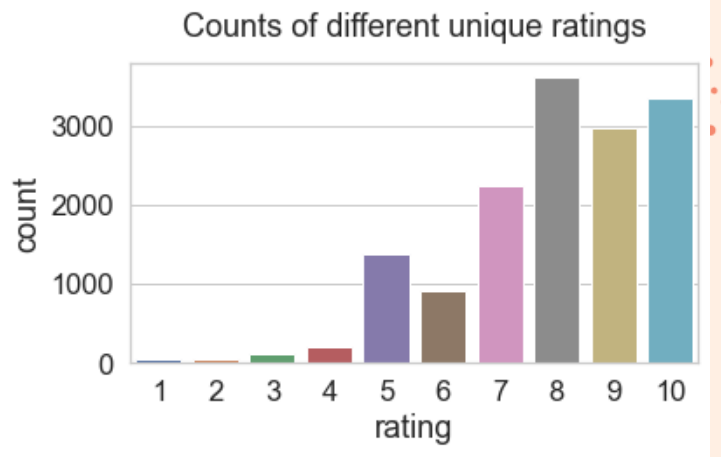
Explore datasets:

- The books dataset is containing 271360 entries, 5 columns
- The user's dataset is containing 278858 entries, 3 columns
- Ratings dataset is containing 526356 entries, 3 columns

Step 3: Data Visualization



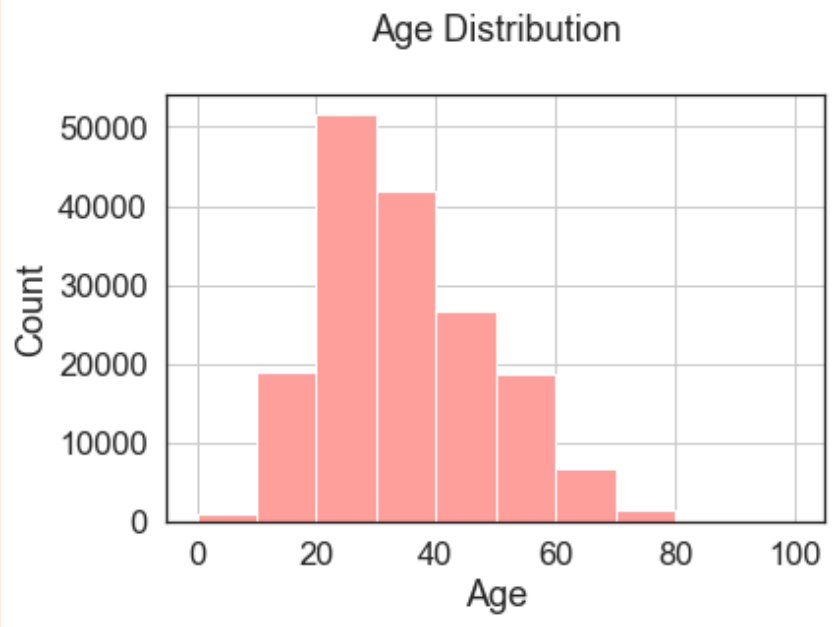
a. Number of book without rating



b. Removing the zeros value from rating

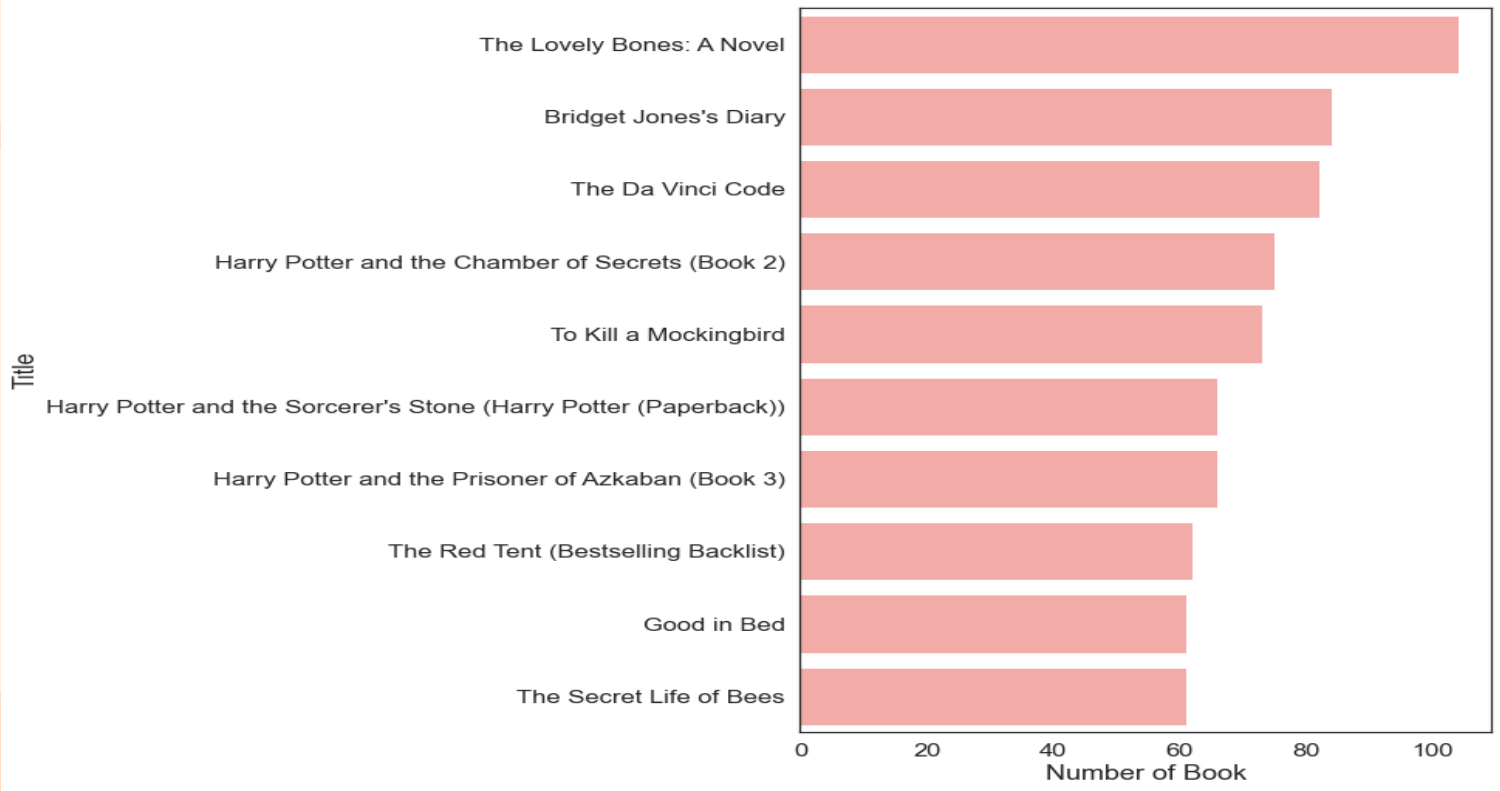
Step 3: Data Visualization

from the diagram we observed the most readers are between 20-30 years



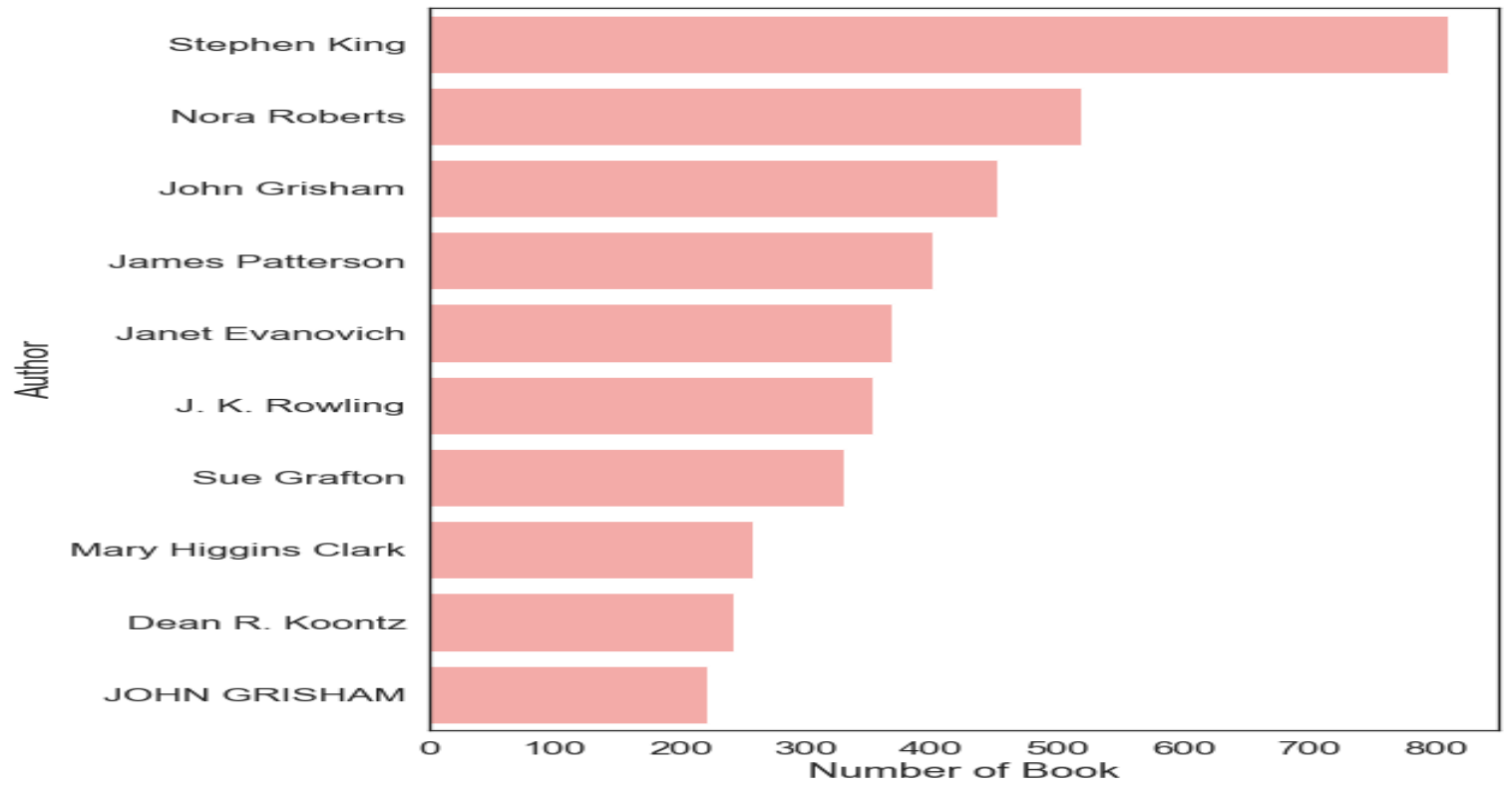
Step 3: Data Visualization

Top books rated by users



Step 3: Data Visualization

The authors per the most number of books



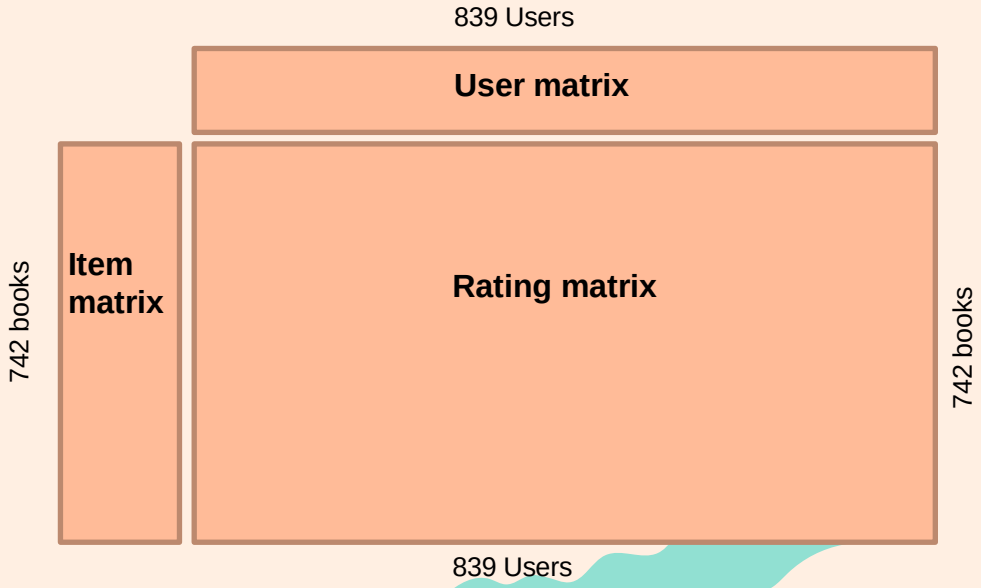
Prepare data for modeling

1. Extract 200 of users and ratings as a features.
2. Merge ratings with books
3. Extract books that have received more than 50 ratings, to reduce the dataset size.
4. Create a pivot table where columns are users and indexes are books and values are ratings.

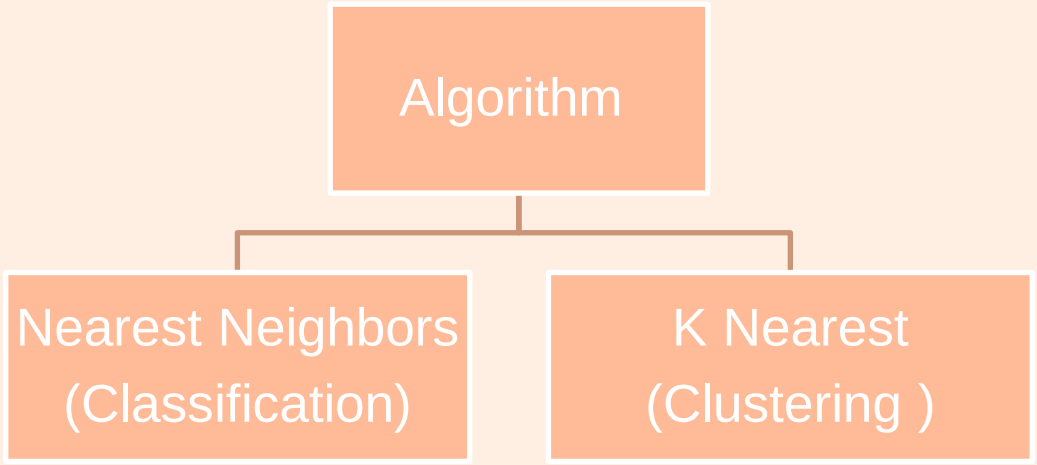
Matrix Factorization

Most famous and widely used Collaborative filtering technique.

In this project, the title, user-id and rating as a matrix, Aim is to find all these user and item dependencies in the matrix.



Step 4: Modeling



Step 4: Modeling

- **Collaborative Filtering**

The system generates recommendations using only information about rating profiles for different users or items. By locating peer users/items with a rating history similar to the current user or item, they generate recommendations using this neighborhood.

- **The k-nearest neighbors (KNN) algorithm**

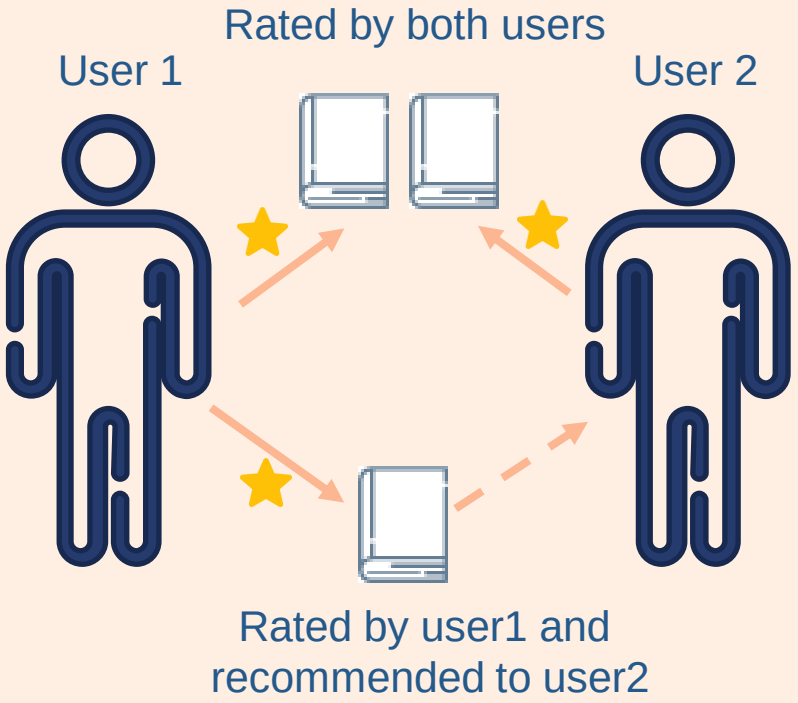
Supervised machine learning algorithm used to solve both classification and regression problems.

- **kneighbors**

Unsupervised nearest neighbors learning which is used for clustring problem.

Step 5: recommendation system

How it’s work?



03

Result

Recommendation system: user-based collaborative filtering

User_id

Book title	user1	user2	user3
Harry Potter and the Sorcerer's Stone	7	9	8
Harry Potter and the Chamber of Secrets	8	4	6
Harry Potter and the Prisoner of Azkaban	0	7	5

Recommendation system result

The system will predict by finding the co-ratings between users, by entering the book ID in the first model(KNN) then we pass the result into the second model(KN) to make suggestions of the 5 closest books.

```
Index(['Harry Potter and the Prisoner of Azkaban (Book 3)',
       'Harry Potter and the Goblet of Fire (Book 4)',
       'Harry Potter and the Chamber of Secrets (Book 2)',
       'Harry Potter and the Sorcerer's Stone (Book 1)',
       'Harry Potter and the Order of the Phoenix (Book 5)'],
      dtype='object', name='title')
```

Tools

04

Tools

1 

2 

3 

4 

Conclusion

05

Conclusion

Finally, I built a user similarities-based recommendation system and found that books could be recommended to users without reference to private information, such as users' interests. Only the user's evaluation of the book can be satisfied, and this method is very effective in terms of time and effort from the user. The disadvantage of this method is that the user may not have similarities with other users, so it is better to combine the types of the recommendation system for diversity and give better results.

Thanks!

Do you have any questions?