

Recommendation Systems

Problem Statement

With the high use of the internet today, we have seen an increase in the number of things people do on the internet. This includes watching movies, reading books, shopping, etc. Because of the wide range of products available to the user, it can be overwhelming to select a preferred item. To overcome this problem, recommendation systems are built to help users find items that have been considered similar by the algorithm to items previously read or items from other similar users. (Anand Shanker Tewari & Kumari Priyanka, 2014)

This paper walks you through the technique used to recommend the most suitable books to user according to books read by a similar user and ratings given to a similar book. This can also be termed content-based and collaborative-based recommendation systems. We will be using Object Oriented Programming concepts to build our system.

Implementation Decisions

The project was built using the Object-Oriented Programming design concept written using python programming language. In python, we used the different libraries available, some of which include seaborn, matplotlib for graphs and visualization, pandas for data manipulation, etc.

There are different approaches to building a recommendation system. These approaches depend on whether we want to get recommendations based on similar users(collaborative) or based on similar books(content-based) or a mixture of both. For this project, we will be looking briefly at only content-based and collaborative-based.

Collaborative filtering (recommendation systems) is based on the assumption that if a user has read a book and rated it, and another user has read one of these books and rated it alike (same or similar score), then the other book the other user has not read can be recommended. (Avi Rana, K. Deeba, 2019)

Content-Based Recommender systems try to recommend books or items based on an item the user has interacted with before. This assumes that if a user read book and gave it a high rating, then it will find other books like book a. This is done by checking the ratings given to a by users, getting books these users have read that have the same (or similar) rating and recommending it back to the user. (Pasquale Lops, Marco de Gemmis and Giovanni Semeraro, 2011)

Justification

Object Oriented Programming is a design model that enables software design around objects(data) instead of functions. It places its focus on the data objects that developers want to manipulate rather than the logic necessary to manipulate them. This method is best used with large systems that are updated at intervals and complex. (Alexander S. Gillis, Sarah Lewis, 2021)

Recommender systems are large systems that involve the use of a dataset that runs into millions of observations. Also, these systems need to be updated/maintained as often as possible. For a system as large as this, using OOP makes it easier to develop and maintain. Concepts in OOP like abstraction, inheritance, polymorphism reduce redundancy in coding and improves efficiency.

Class Diagram

The class diagram is used to show the overview architecture of the system. The interaction between classes are shown in the image below.

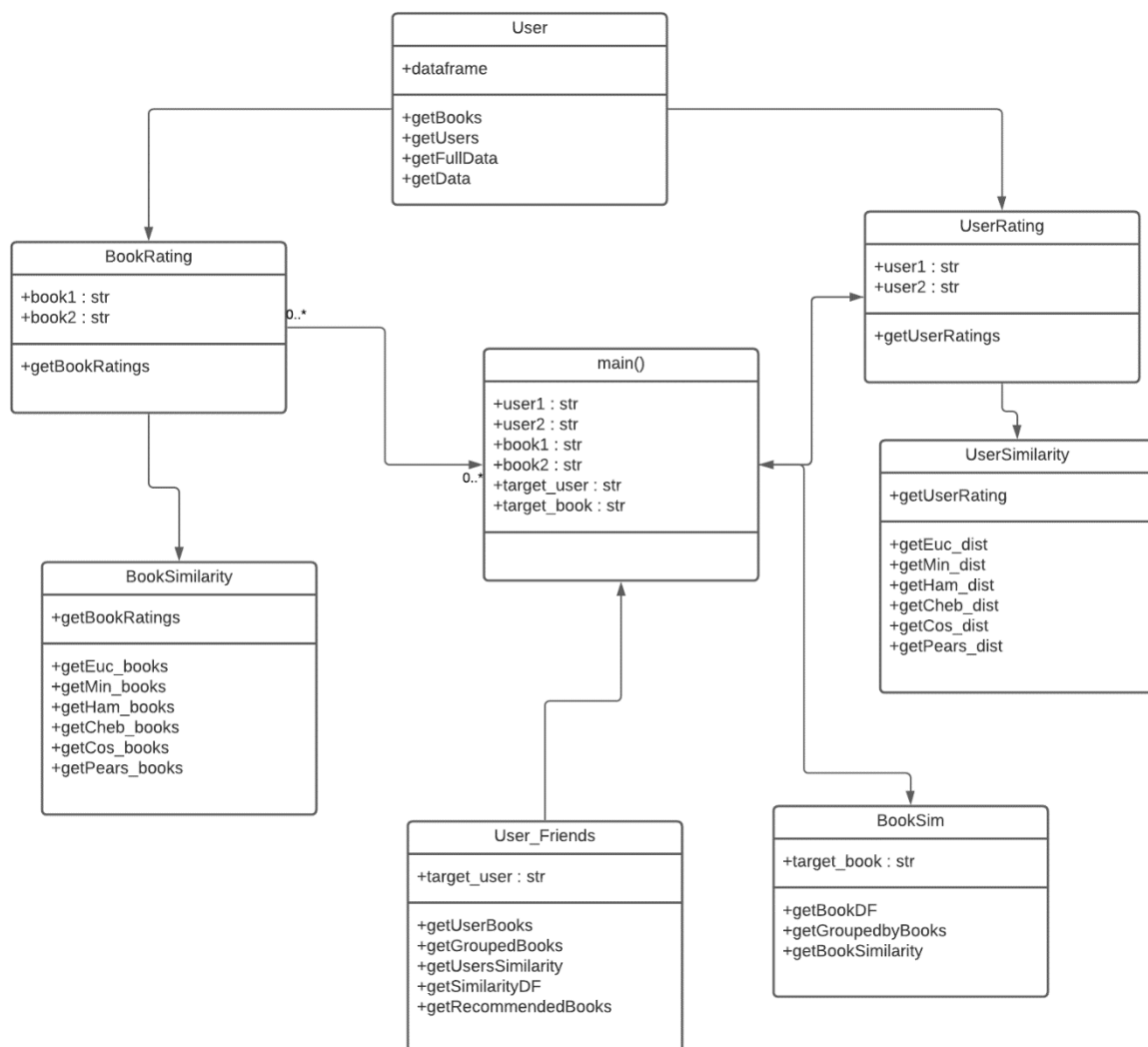


Figure 1

Architectural Diagram

Architectural diagrams are used to show the working routine of a software system and the relationship between its components. The figure below shows the architectural diagram of a recommendation system as produced by (Avi Rana, K. Deeba, 2019)

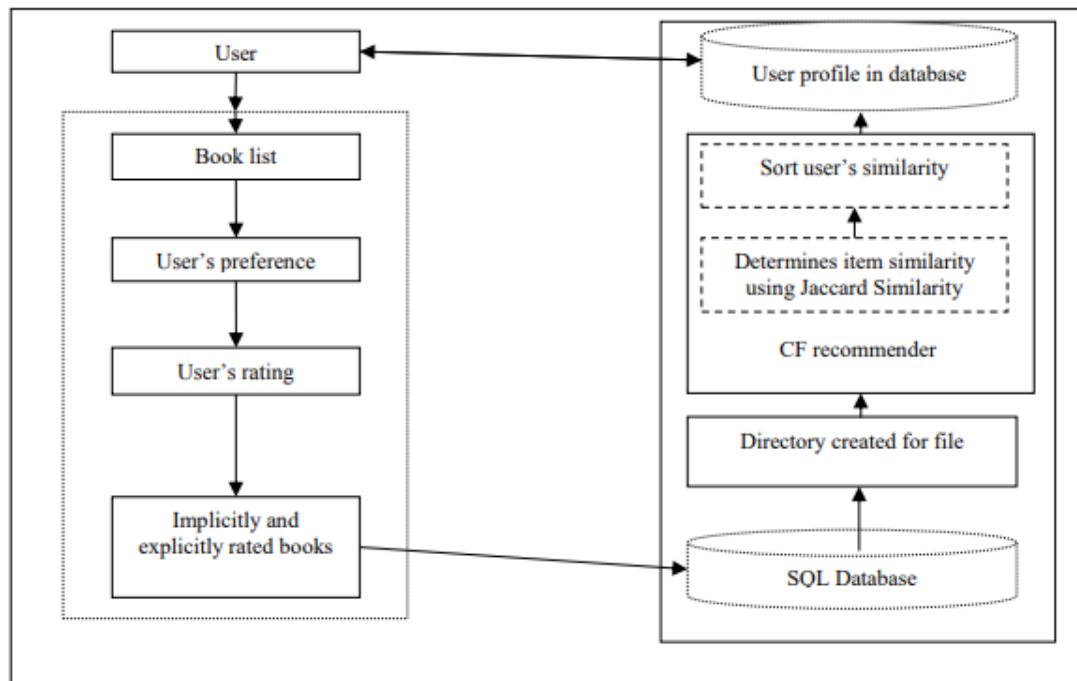


Figure 2

Flow Diagram

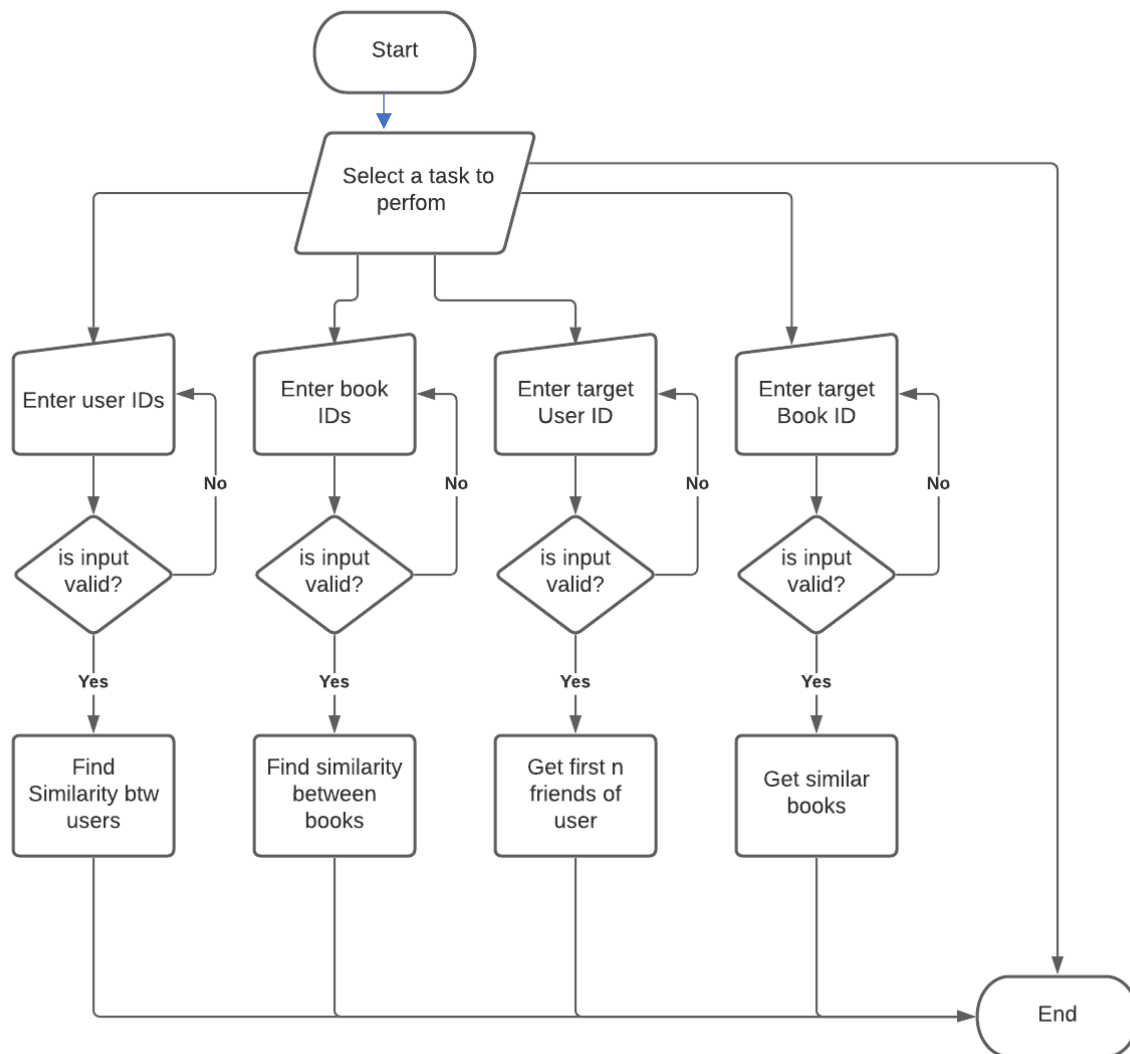


Figure 3

Reflection

From the first project done on recommendation systems, I was able to learn a lot about loading datasets and putting implementation into modules. This project came with its own struggles and most importantly learning experience. I struggled mostly with manipulating data frames but with each challenge, I learnt something new. Loading datasets directly into the data frame was fairly the easiest thing to do. With the help of Stack Overflow, Pandas Library, GeeksforGeeks and the internet, I was able to navigate my way through a lot of the problems with a little help from colleagues. My biggest challenge is transitioning into machine learning, which is the KNN aspect which I was unable to implement as I could not understand it early enough. Even though I know how to use matplotlib, I have not added any graphs yet as it was only recommended after implementing KNN. In all, it has been an interesting experience as I have learnt beyond what I thought I could, and I am grateful for this. I intend to practice more so I can get better with python and machine learning.

References

- Alexander S. Gillis, Sarah Lewis. (2021, July). *Search App Architecture*. Retrieved from Tech Target: <https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-OOP>
- Anand Shanker Tewari & Kumari Priyanka. (2014). Book Recommendation System Based on Collaborative Filtering and Association Rule Mining. *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, 1-4.
- Avi Rana, K. Deeba. (2019). Online Book Recommendation System using. *Journal of Physics: Conference Series Collaborative Filtering (With Jaccard Similarity)* , 1.
- Pasquale Lops, Marco de Gemmis and Giovanni Semeraro. (2011). Content-based recommender systems: State of the art and trends. *Recommender Systems Handbook*, 2-5.