**PROJECT IMPLEMENTATION**

**Book recommender system**

# Collaborative Filtering

# 

Collaborative Filtering is a recommendation technique used to create a recommendation algorithm based on the preferences of users or items (products, content, etc.). It is commonly employed in various domains such as e-commerce websites, movie and TV show platforms, music applications, and more.

The principle behind collaborative filtering is to recommend items to a user by identifying other users or items with similar preferences, based on their past choices. In other words, it tries to find users who have similar tastes and recommend items liked by those similar users.

There are two main types of collaborative filtering approaches:

* **User-Based Collaborative Filtering:** In this method, the preferences of a particular user are compared with the preferences of other users to find those with similar interests. Then, items liked by these similar users are recommended to the target user.
* **Item-Based Collaborative Filtering:** In this approach, the liking patterns of users for specific items are compared to identify similar items. Items similar to the ones the user has liked are then recommended to the target user.

About Dataset

Context

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. As a proof of the importance of recommender systems, we can mention that, a few years ago, Netflix organised a challenges (the “Netflix prize”) where the goal was to produce a recommender system that performs better than its own algorithm with a prize of 1 million dollars to win.

By applying this simple dataset and related tasks and notebooks , we will evolutionary go through different paradigms of recommender algorithms . For each of them, we will present how they work, describe their theoretical basis and discuss their strengths and weaknesses.

Content

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 case of several authors, only the first is provided. URLs linking to cover images are also given, appearing in three different flavours (Image-URL-S, Image-URL-M, Image-URL-L), i.e., small, medium, large. These URLs point to the Amazon web site.

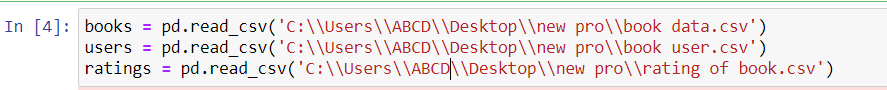
* **Ratings**

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.

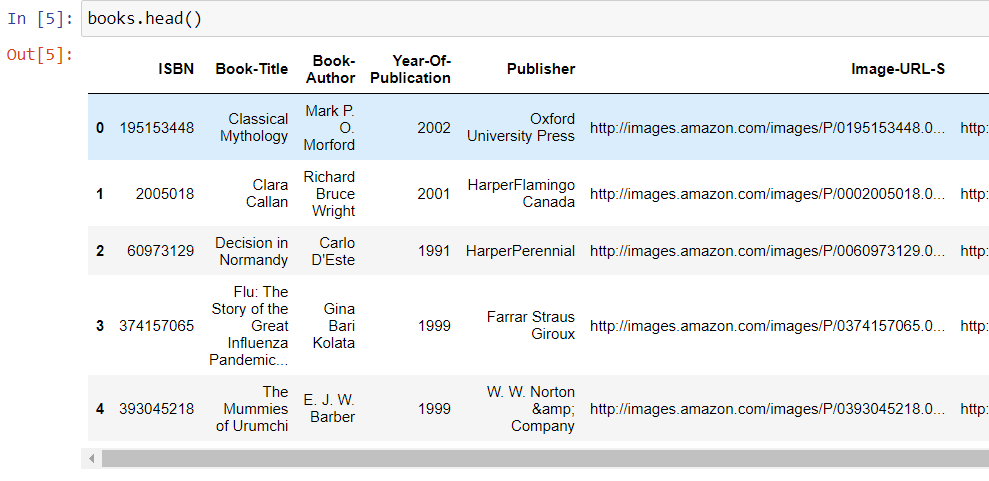
## **Libraries and Utilities**



## **Loading the dataset**



# Dataset Overview



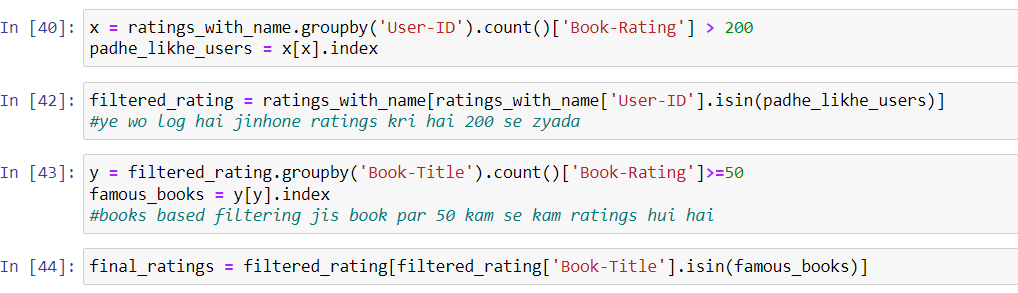
Our criteria for accuracy

We know that we have a sizable dataset, which may also contain inaccurate data, but since a recommendation system requires accuracy, we create a set of criteria.

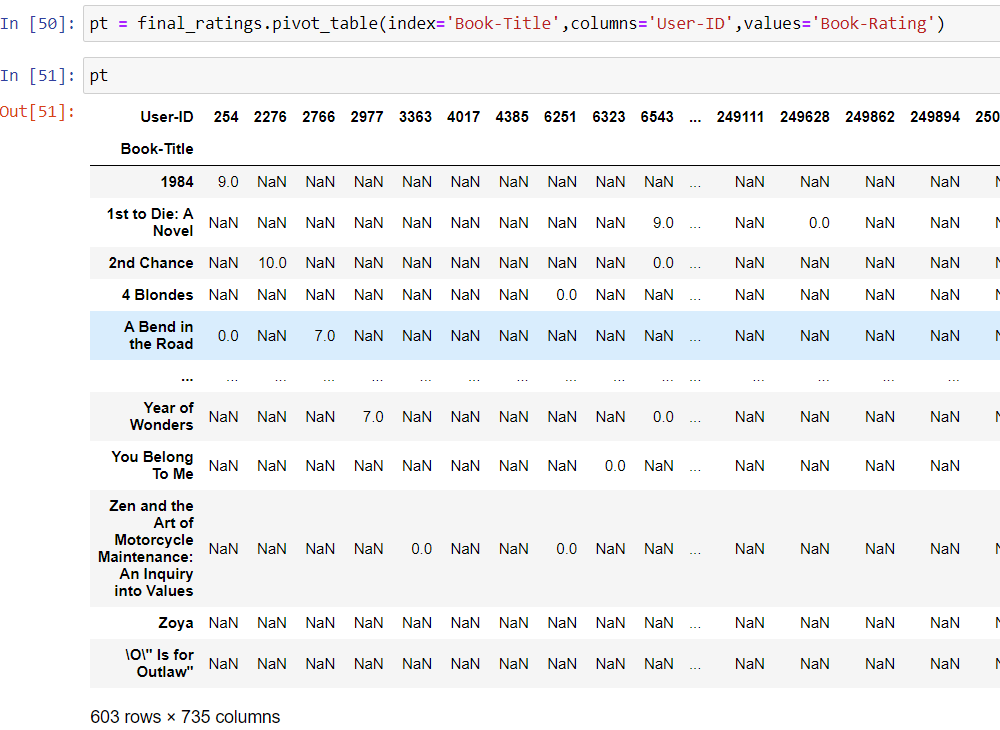
1. the quantity of books with at least 200 user ratings.

2. The user rates a minimum of 50 books.

We are constrained by our data since, based on the aforementioned criteria, we only have data sets that meet our criteria.



Pivot tables

A pivot table is a similar operation that is commonly seen in spreadsheets and other programs that operate on tabular data. The pivot table **takes simple column-wise data as input, and groups the entries into a two-dimensional table that provides a multidimensional summarization of the data**. 

Our similarity scores and our recommend function/algorithm



Check whether our recommender works or not



From the above result, we can see that our recommendation system works; it gives us the book name, book author, and book image URL, which are required.

**Now we want to show this on the web page**.

For this, we need a pickle file, or as a result, we dump all the pickles in Jupyter and use them further in the web page.

**How to show in web page?**

**We are using python flask , html , bootstrap to show our result in web page.**

# What is Flask Python

Flask is a web framework, it’s a Python module that lets you develop web applications easily. It’s has a small and easy-to-extend core: it’s a microframework that doesn’t include an ORM (Object Relational Manager) or such features.

It does have many cool features like url routing, template engine. It is a WSGI web app framework.

**In our project appp.py is the flask .**