**Book rating prediction using Collaborative Filtering**

**Team members:**

Anqi Chen

Xinhe Chen

Yisu Tian

**Introduction:**

Collaborative filtering is one of the most effective methods to build a recommender system. Our group will use this method to recommend future reads to the user based on their ratings for books they have read. Item based collaborative filtering will be used to create an item-user matrix. Unrated items for each user will be predicted using our algorithm.

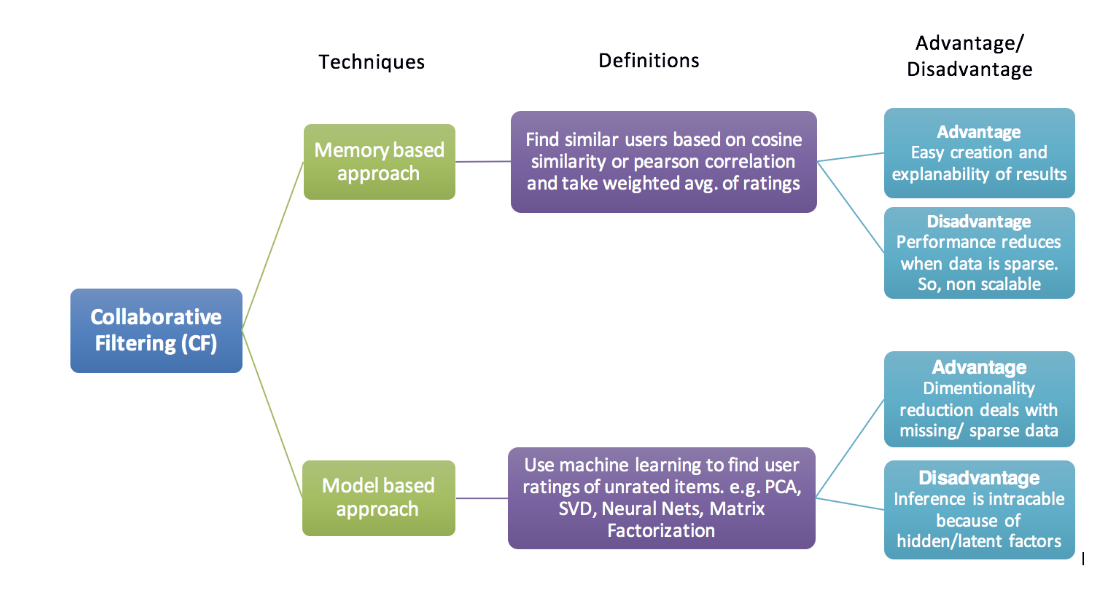
**Problem Definition and Algorithm:**

**Task Definition**

A recommender system recommends users items based on items they have rated. Ratings for ten thousand books have been collected and a recommender system

**Algorithm Definition**

There is generally two approaches to collaborative filtering. Memory based and model based as shown in figure 1. Memory based approach uses cosine similarity to calculate the ‘distance’ between users or items and take a weighted average of ratings. It is easier to create than the model based approach but suffers when there is not enough data. Model based approach uses methods such as neural networks and does not need as much data, however it is harder to implement. This project will use the memory based approach as there is sufficient data present.

Figure 1. Types of collaborative filtering approaches [2]

Memory based approach can be further separated into two categories. User-Item collaborative filtering and Item-Item collaborative filtering. User-Item filtering takes a user and find similar users and recommend the items these similar users like. Item-Item filtering takes an item a user likes and finds users who also like this item and recommend the items these other users like.

**Experimental Evaluation**

This project uses Python 2.7 with libraries including numpy, pandas, sklearn.

The analysis will use memory based collaborative filtering to predict the ratings of books for users based on their ratings for other books.

The data used contains ratings for ten thousand books. With more than fifty thousand users, each user has at least made 2 ratings. The data will be separated into training data and testing data to evaluate the algorithm. ¾ of the data will be fed to the algorithm as training data and ¼ of the data will be testing data. Due to the large data present, regular computers can run into memory issues when processing the data. The matrix may need to be shortened/cut to make the program runnable.

<https://github.com/zygmuntz/goodbooks-10k>

**Results**

**References**

[1]<https://blog.dominodatalab.com/recommender-systems-collaborative-filtering/>

[2] https://towardsdatascience.com/various-implementations-of-collaborative-filtering-100385c6dfe0