Documentations Reading Sequence:

Please read the documents in the following order from the submitted folder to understand the whole idea and process of this project.

1. ReadMe.pdf
2. Preprocessing (MySQL).pdf
3. TF_IDF (Python).pdf
4. Recommendation (Neo4j).pdf

Brief Idea:

A wedding registry is a website where the engaged couple put their gift preferences to the wedding guests. Therefore, the couple may want to get a recommendation by utilizing the gift preferences of the wedding registries of other similar married couples. Again, the guests, when selecting a gift from the wedding registry, can be suggested a few gifts that they might like to buy for the couple.

My Scenario:

Let the engaged couple happen to be very fond of books. They would like to get a recommendation to put 100 new books on their wedding registry that they never read before. This recommendation can be provided by the other 100 married people who are similar to the couple and rated those unread (by the couple) books.
Let the engaged couple invited 100 of their friends to the wedding. To save time on looking over all the 100 books in the wedding registry, those invited friends would like to get a recommendation of only a few books from the registry so that they can make the gift selection easily. This recommendation can be made by utilizing the friends' rating history to many books and the books' features.

Dataset:

The dataset consists of ‘Users’, ‘Books’, and ‘Ratings’ entities.
There is a total of 202 users. 2 of them (id 12874 and 30944) are considered as the 'couple'. 100 of them are considered 'Married' people and the remaining 100 of them are considered as invited friends.
There is a total of 5056 books. Books also have different properties (title, authors, publication year, language).
And there is a total of 38162 ratings of users to books.
You can find the dataset source and data preprocessing related information in the “2. Preprocessing (MySQL).pdf” document.

Recommendation 1 (Wedding Registry):

To recommend books for the wedding registry of the couple, User Collaborative Recommendation System has been considered. Three similarity methods have been applied to find the similarity of users of type 'Couple' with the users of type 'Married'.

i) Cosine Similarity = $\frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} x_i^2}}$

ii) Pearson Similarity = $\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}$

iii) Mean Square Error $= \frac{\sum_{i=1}^{n}(x_i-y_i)^2}{n}$

Hence, Mean Square Similarity $= \frac{n}{\sum_{i=1}^{n}(x_i-y_i)^2}$

Here, i is a book that both users x and y rated. x_i is x's rating for book i, y_i is y's rating for book i.

Now, for a book that Couple 'C' didn't rate, but Married user 'M' rated, the similairty of that book with 'C' is calculated as

$$\sum_{for\ all\ M} (\text{Sum of the similarity value of M with fiance \& finacee} * \text{M's rating for that book})$$

This is done for all the books that 'C' didn't rate, but 'M' rated.

Finally, I put only the 100 most similar books with the 'C' in the wedding registry. These 100 books might be a little different for different similarity methods used. You can find the details of this process in the 2nd part of "4.Recommendation (Neo4j).pdf" document.

Recommendation 2 (Friend Selecting Gift):

Content-based Recommendation system has been used to find the books which are most similar to the invited friends. Hence, users of type 'Friend's rating history and the books' property have been used. TF_IDF (Term Frequency, Inverse Document Frequency) method has been used to calculate the numerical values (relative frequencies) for each word of the book title, each author, each publication year, and each language found in all the books in the dataset. Then the users of type 'Friend's similarity with each of these features (words of the book titles, authors, years, and languages) of the books are calculated. You can find the process of TF_IDF over the book data and User Profile creation in the "3.TF_IDF (Python).pdf" document.

Now for each user of type 'Friend,' I consider the 100 books in the wedding registry and calculate the books similarity with the users based on all the features (title, author, year, language) of the book and suggest only the top 10 books for that 'Friend' to gift to the couple. Based on the different sets of the books in the registry (calculated by Cosine, Pearson, and Mean

Square similarity method), the gift suggestion to the invited friends might be a little different. You can find the details of the process in the 3rd part of “4.Recommendation (Neo4j).pdf” document.

Results:

You can find the findal results of different output in the xlsx files in the “4. Recommendation (Neo4j)/results” folder.