HW7: Collaborative Filtering_ Amazon Book Recommendation System

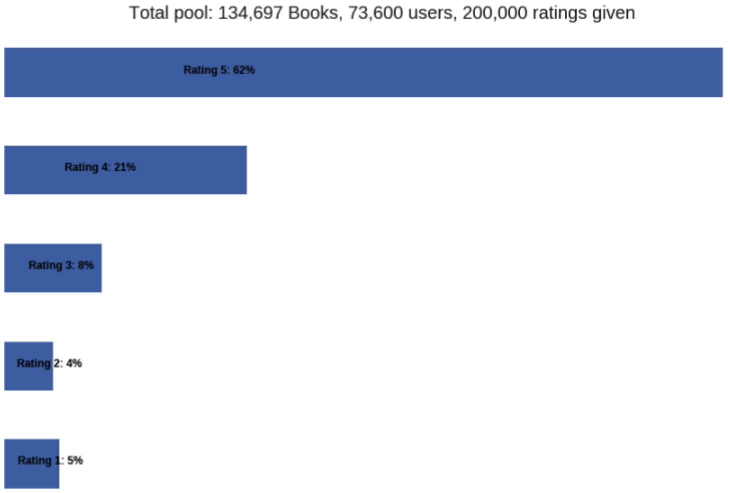
Overview

With the rapid growth of data collection, more efficient systems are created based on the big data. Recommendation system is one of information filtering systems which could improve the quality of search results and provide items that are more relevant to the search item or are related to the search history of the user. Almost every major tech company has applied recommendation systems in some form or the other: YouTube uses it to decide which video to play next on autoplay; Spotify uses it to provide users "Made for you" daily mixes etc.

In terms of this project, we try to learn from data and recommend best books to users, based on self & others behavior. This dataset contains book reviews from Amazon, with columns User ID, Book ID, Rating (1 to 5) and Date they gave the ratings (timestamp Unix). Because of the huge data size, the ratings_Books_sample.csv used is sampled from the original dataset with 200K records.

Technique

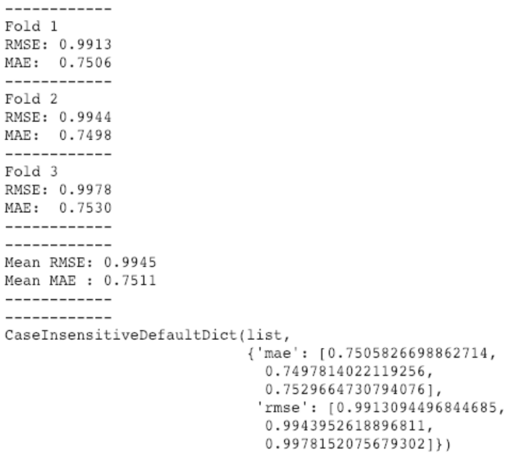
In the total pool, there are 134,697 books and 73,600 users. Firstly, the bar chart is made to give us a first look on how the data spread. We can see that the rating tends to be relatively positive (83% of the ratings > = 4), which may be due to the fact that unhappy readers tend to just leave instead of making efforts to rate. As a result, it indicates that low rating books are generally very bad.



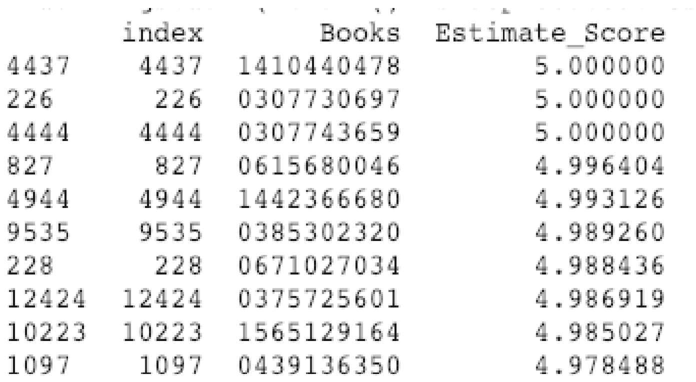
Before building the model, to improve the data quality, the following two steps are done: remove books with too less reviews (they are relatively not popular); remove users who give too less reviews (they are relatively less active). After trimming down the data, the data size changes from 200K records to 130K.

To create collaborative filtering recommendations, Surprise library is utilized here to implement SVD. Collaborative filtering matches persons with similar interests and provides recommendations based on this matching. Collaborative filters do not require item metadata like its content-based counterparts.

The result of 3-fold RMSE, MAE of algorithm SVD is as follows. We get a mean Root Mean Square Error of 0.99 approx. Then, we could use the model to predict one specific user's preferences. For instance, we focus on the user A00006923FEAFJLE7GHEL and the top 10 books he or she would love to read are shown below based on the model estimation.



```
------------
Fold 1
RMSE: 0.9913
MAE:  0.7506
------------
Fold 2
RMSE: 0.9944
MAE:  0.7498
------------
Fold 3
RMSE: 0.9978
MAE:  0.7530
------------
------------
Mean RMSE: 0.9945
Mean MAE : 0.7511
------------
------------
CaseInsensitiveDefaultDict(list,
                           {'mae': [0.7505826698862714,
                             0.7497814022119256,
                             0.7529664730794076],
                            'rmse': [0.9913094496844685,
                             0.9943952618896811,
                             0.9978152075679302]})
```



```
       index       Books  Estimate_Score
4437    4437  1410440478        5.000000
226      226  0307730697        5.000000
4444    4444  0307743659        5.000000
827      827  0615680046        4.996404
4944    4944  1442366680        4.993126
9535    9535  0385302320        4.989260
228      228  0671027034        4.988436
12424  12424  0375725601        4.986919
10223  10223  1565129164        4.985027
1097    1097  0439136350        4.978488
```

Conclusion

Based on Amazon book reviews data, SVD is implemented to create the collaborative filtering recommendation and such recommendation system could estimate book ratings for users and recommend 10 books a user would love to read which have the highest estimated ratings.

However, one of the limitations of this project comes from the sampled Amazon book review data. Because the original dataset is too big, I just sampled 200K records from it. As a result, there are a large number of books which do not have many reviews. (Remember although I have removed books with too less reviews using 0.8 quantile, the book minimum times of review are still 1.) The data quality would influence the accuracy and efficiency of the recommendation system to some degree. If we could find a dataset within which each book has many reviews and each user rates many books, the model is expected to be better. In addition, the dataset only has book id but does not have the corresponding book names, which increases the difficulty for us to check the recommendation results and get insights from it.

Link

https://colab.research.google.com/drive/1_x4VqLTR7cXNk8oAhJHz2Fk49R580dRo