Title: Recommender System assessed exercise report

Name: Chentao Xu

ID: 2438322X

In the first question, I started from the non-personalized collaborative filtering algorithm and calculated the four variables in the question respectively. Since the non-personalized collaborative filtering does not take into account the preferences of each user, the results obtained are only the objective parameters of each book.

The first approach is to average the ratings for each book and use the average as a predictive score for each book, allowing the system to make a recommendation because each book will have a different average score between 0 and 5.

The second method is to obtain the total number of ratings of each book from the table, including all the scores of 1-5 for these books scored by the user, and use the number of scores as the basis of the system's recommendation for all users.

The third method is to get the number of times each book is rated by the user as a 5, which means that the more books the user scores with a 5, the more priority the system should give to the recommendation.

The last method is a transformation of the third method, because the total number of times which each book is rated are different, the total number of times the book is rated is not enough to prove that the book is the most popular. Therefore, in this method, we use the ratio of five points for books as the basis of the system's recommendation priority.

After calculating the four variables, we need to evaluate the four models. In this experiment, the main method to evaluate the four models is mean-rank, which refers to the Mean of the reciprocal ranking of multiple query statements. All four models used explicit data sets as training sets and were evaluated using test sets to get an average score for all users. The higher the average of the MRR shows that the higher the accuracy of the model, so we can get the third method model is the best, because this model is made by the user to forecast the number of books play out, because the user play out the more that have seen the book user satisfaction, the higher of the book. However, the fourth model has the lowest score, the reason might be that proportion limit features to some degree.

```
The MRR score for average model is: 0.015052024168984034
The MRR score for rating num model is: 0.2396001188245477
The MRR score for rating_5 model is: 0.2409670879930144
The MRR score for fraction_5 model is: 0.03415267465103555
```

Fig 1: The MRR score of four models in question 1

In the second question, I first used explicit factorization model and implicit factorization model to train explicit data sets respectively, tested them on the test set, and evaluated the training results of the two models with MRR algorithm. The results show that the implicit factorization model has a higher score, which means that the books recommended by the model are more accurate. Explicit data sets contain data sets that users clearly rate, which can truly reflect how much users like books. In explicit model, the inner product mainly fits the users' actual scores for various books, and only fits the books with scores. However, in implicit model, it mainly fits the users' preference for articles, so the implicit model performs better based on fitting explicit dataset.

Then, we need to select the best embedding dimension parameter in implicit factorization model to select the best model. This process mainly uses the validation set separated from the training set to continuously verify and adjust the super parameter. Therefore, we can see from the results that the MRR scores of the implicit models under four different embedding dimensions are very close, and the training effect of the model is the best when the dimension is 8.

```
The MRR score of implicit model with latent factor 8 on explicit data is : 0.0159385938742
The MRR score of implicit model with latent factor 16 on explicit data is : 0.016254096382
The MRR score of implicit model with latent factor 32 on explicit data is : 0.014930964036
The MRR score of implicit model with latent factor 64 on explicit data is : 0.014370484397
```

Fig 2: The MRR score of four models in question 2

In the third question, I still made a training fit for implicit model, but this time the training set was a recessive data set. Implicit data sets contain books that users have viewed or want to view, which means that we cannot judge users' clear attitude from these data, so this kind of data can only be regarded as books with positive feedback. Moreover, the noise data of the hidden data set will affect the training results of the model, but its large amount is another advantage. The training results of the models with four different embedding dimensions are still close, but when the test sets are all validation sets, the implicit data set is indeed much better than the explicit data set for model cooperation. Similarly, we still chose 16 as the embedding dimension in this super-parameter selection, because its MRR score was the highest. Therefore, we can think that the large number of recessive data sets can mask the shortcomings of noise data to some extent, because I print the real number of books which the user really like and the number of books which the model predicts correctly.

```
------------------------------------------------------------
User with id 1591 with RR 1.000000 had 91 ratings
The user 1591 actually put 24 books on the shelves!
The model predicts  4  books correctly in top rank 30 books!
------------------------------------------------------------
User with id 1598 with RR 1.000000 had 30 ratings
The user 1598 actually put 17 books on the shelves!
The model predicts  2  books correctly in top rank 30 books!
------------------------------------------------------------
User with id 1608 with RR 1.000000 had 72 ratings
The user 1608 actually put 10 books on the shelves!
The model predicts  4  books correctly in top rank 30 books!
------------------------------------------------------------
User with id 1624 with RR 1.000000 had 106 ratings
The user 1624 actually put 21 books on the shelves!
The model predicts  1  books correctly in top rank 30 books!
------------------------------------------------------------
User with id 1635 with RR 1.000000 had 88 ratings
The user 1635 actually put 17 books on the shelves!
The model predicts  2  books correctly in top rank 30 books!
------------------------------------------------------------
```

Fig 3: The part of data about correctly predicting

Intra-list score represents the similarity between two items. The higher the score, the further apart the two items are, and the lower the similarity. Therefore, we used the best model in the third problem to predict the users with the highest score and the users with the lowest score. It is obvious from the results that the book similarity of the predicted results obtained by the users with high scores is very small, while the book similarity of the predicted results obtained by the users with low scores is very high, with the keywords of ‘Harry potter’.

```
17    Harry Potter and the Prisoner of Azkaban
Name: original_title, dtype: object
1    Harry Potter and the Philosopher's Stone
Name: original_title, dtype: object
22    Harry Potter and the Chamber of Secrets
Name: original_title, dtype: object
23    Harry Potter and the Goblet of Fire
Name: original_title, dtype: object
24    Harry Potter and the Deathly Hallows
Name: original_title, dtype: object
23    Harry Potter and the Goblet of Fire
Name: original_title, dtype: object
125    Dune
Name: original_title, dtype: object
639    Definitely Dead (Sookie Stackhouse, #6)
Name: original_title, dtype: object
1204    Children of Dune
Name: original_title, dtype: object
192    Outliers: The Story of Success
Name: original_title, dtype: object
```

Fig 4: The five books’ names in low and high intra-list score respectively

In the fourth problem, we used the unweighted hybrid recommendation model, and combined the best model in the second problem with the best model in the third problem, taking the union of the results predicted by the two models as the predicted value of the unweighted hybrid model. Then, the new hybrid model was used to evaluate the model in the test set, again using the MRR algorithm to evaluate the score. Finally, by comparing the scores of all users obtained by this new model in the test set with the test results of the other two models, it can be found that the new mixed model is more accurate in predicting the results of users, because most users' scores are close to 1. In addition, after printing the number of users whose RR score improved after combination, I found that there are 345 users' RR scores improved after combination which are higher than the best model in question 2 and question 3.

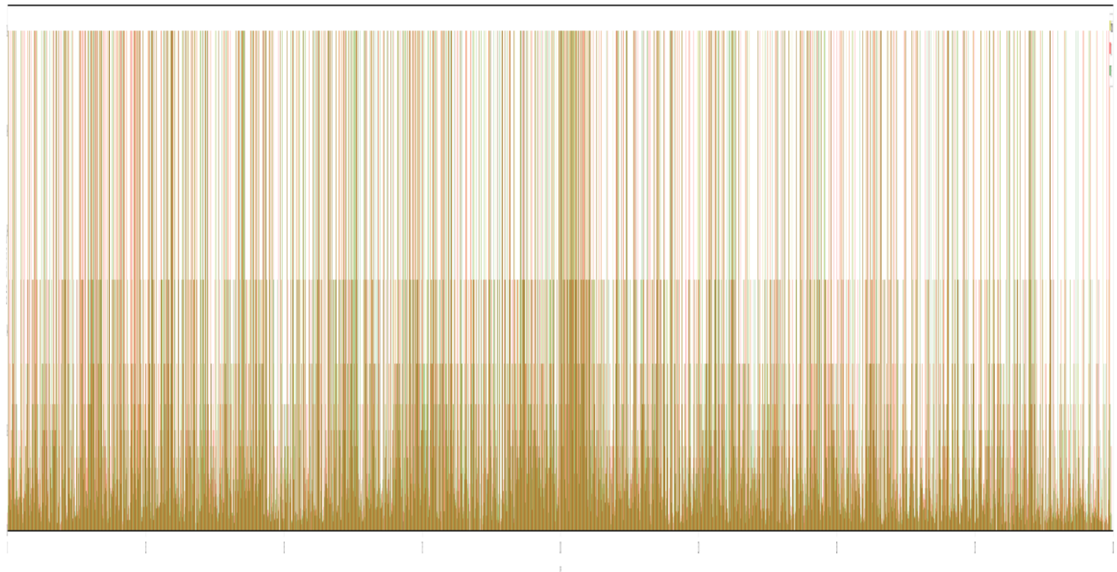

Fig 5: The RR score among combination model, best model in question2 and 3

In the fifth problem, we use Lift, which is used to judge the degree of interaction between the two books. Firstly, I screened the original table and selected the part of the book whose ratings were greater than or equal to four points. In my offline model, I firstly store all valid book pairs as dictionary keys and the lift value of each pair as value in a dictionary. Then I loop through the ids of all the books and determine if they have been overtyped by the user. If it is a book that the user has already rated, set the prediction to 0, or it will go to the next loop. The next loop loops through all books that a single user has scored more than four points. If the book pair does not appear in the previous dictionary, the loop continues. If the book pair appears in the previous dictionary, log the corresponding lift value and sum it. Finally, I used a map to map the book ids and sort them by value size. And the MRR score for my offline model is 0.36. In the online model, I use the first user to predict the final score.

I use a dichotomy when calculating the optimal threshold. First of all, I took the maximum number of times in the book as the upper limit, and kept using dichotomy to calculate MRR scores under different thresholds. After doing four dichotomies, I got the optimal threshold of about 65.

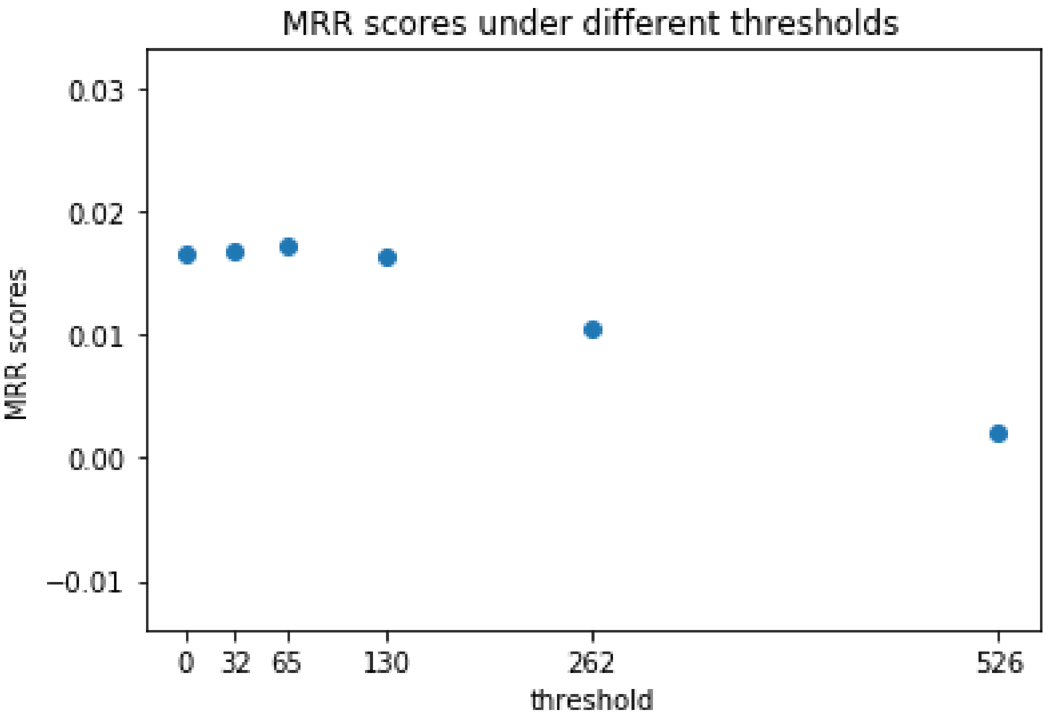

Fig 6: The MRR score among different thresholds

In question 6, there are four combinations, and the model I chose in question 1 is a Fraction model. In question 2, I chose the best recessive model. In the four combinations, since the sum of the weights must be 1, I used a nested loop to calculate the weights in different cases. Unfortunately, I found this run time to get very long by printing every time I tried, because the computation was so complex, so I changed my approach and used a combination of random weights and tested on the validation dataset. Since the combined prediction efficiency of the models in question 1, question 2 and question 3 is relatively high, I have obtained the three weights when the MRR score is the highest by means of circulation, respectively 0.6, 0.1 and 0.3. Therefore, I put the values of these three weights into the model and calculated the MRR score (0.2448) on the test dataset. Then, I combined the models in question 2, question 3 and question 5 and calculated the MRR score using random weight values. And the final MRR score for combination of question 2, 3 and 5 best model is 0.39 which is higher than the MRR score of lift model.

In the seventh question, combining all the previous experimental results, first of all, I think efficiency and effectiveness should be considered when choosing the recommendation system, because only by combining the two aspects of evaluation can the best recommendation system be selected. From the above experimental results, it can be seen that the recommended effect is the best after combining the optimal implicit factorization model and the optimal factorization model to train the data set, and the running time is very short, but there are some contingencies, for example, the implicit matrix model is more tolerant of noise data.

However, when I add different weights to the combination of different models in question 6, the effectiveness improved compared with the previous single model prediction, but the efficiency is really low. Therefore, I suggest to use the validation set to select the best parameters of different models firstly, but the models need to choose the ones with higher operation efficiency. In the comprehensive experiment, I will choose the explicit factorization model and the implicit factorization model, because they are much more efficient than the lift model, and the MRR scores are also close. Then the two models are combined and different weights are added to fit until the best MRR score is obtained. Secondly, I think some mathematical skills can be added to reduce the computational complexity when aiming at the implicit factorization model, in order to further improve the efficiency and meet the demand of online computing. Finally, I believe that when fitting the implicit data, we need to perform some operations on the data set, such as centralization, to reduce the impact of noise data on the model fitting.

After this experiment, I have more understanding of the recommendation system. First of all, I think the key of recommendation system is how to find users' interests and recommend relevant products to users according to users' interests. I think there are many ways to evaluate a recommendation system, but a real recommendation system should use its immediacy as a key evaluation factor. Secondly, I think the recommendation system needs to constantly classify and analyze the commodity information, user information and user behavior, so that we can fit the model to more effective data. In addition, when choosing different models, I believe that while considering efficiency and effectiveness, we should adopt various mathematical models and machine learning methods to continuously improve the efficiency and effectiveness of the model. Finally, in terms of the analysis of the results predicted by various models, I think a lot of classification and visualization are needed for better analysis of the results to improve the accuracy of the recommendation system.

Overall, I think a perfect recommendation system needs to obtain users' long-term preference and short-term preference, which pay more attention to users' preference changes in different time and space. As a result, short-term preference should be used primarily as an auxiliary analysis of long-term preference.