Web Science 2023: Final Project

Taraburca Radu - whx862

Abstract

This document contains the discussions and answers for the final project.

1 Week 6

I calculated the distribution of user and film ratings, as well as average user rating and film. Also, I calculated the first 5 most popular elements based on the number of evaluations. Besides this, I also made the graphs of the distributions of user ratings and movie ratings, as well as the chart of average rating for each film. Based on the data observed after performing statistics on items and users, we can deduce the following:
A first observation that we can easily deduce from charts, but also from statistics is that the data set is quite scatter. On average one user has rated 80 films, but we also have a standard 78.8 deviation, which indicate a great variation in the number of films that each user has evaluated. The minimum number of films evaluated by a user is 4 and the maximum is 635, indicating that some users have specific tastes, while others are more versatile and evaluate more films. Another observation we can make is that some movies are rated more than others. On average a film has 51 ratings with a standard deviation of 66.9, indicating that the number of ratings on film varies a lot. The minimum number of ratings of a film is 1, and the maximum is 484. From the top 5 the most rated films that have at least 390 ratings, we can some films are more popular than others. From here comes the problem of creating a recommendation system that is not a biased. Another easy thing to see is that on average a movie has a rating of 3.12, and the rating range has as a minimum of 1 and maximum 5. The standard deviation of 0.7318 indicates that there is some variability in the movie ratings, some movies being evaluated much higher or lower than the average. From here we deduce that on average the films are rated above average. This is not necessarily a positive thing, because we can assume that users tend to overate movies.

Based on the calculated statistics, the data set has some important properties that should be taken into account during the evaluation.

First of all, because it is such a big difference between the number of users ratings, the minimum being 4 and the maximum of 635, the system could be biased with the more active users' preferences. Secondly, because it is a big difference between the number of film ratings, the minimum number of ratings of a movie being 1 and the maximum of 484, the recommendations could be a biased towards the more popular movies. Lastly, due to the average rating per movie of 3.12 out of 5, there is the problem of over-rating the movies by users. This could impact the accuracy of the recommendation model.

2 Week 7

Since the chosen data set has a small number of movies but also a relatively small number of people, any neighborhood-based model is good to be used, but if we consider the previously mentioned observations, we can see that in this case it is better to use a user-based model. Besides that, another factor that made me choose this model instead of the other one is the large variation in the number of movies that each user has rated.
For the latent factor model, I chose to use the SVD, because the data set contains sparse and noisy data, and this model can deal with these problems.

I chose the RMSE (Root Mean Squared Error) as the assessment metric because it is a simple and straightforward tool for assessing the accuracy of the predicted ratings. I chose it because it is one of the most commonly used metrics for evaluating accuracy, but also because it penalizes larger errors more heavily. Thus, this measure is suitable to measure the accuracy of the predicted ratings.

After 5-fold cross-validation, we obtained the following hyperparameters for the user-based collaborative filtering model:

{'sim_options': {'name': 'msd', 'user_based': True, 'min_support': 5, 'shrinkage': 0}}

This model achieved an RMSE of 0.98 averaged over the 5 folds.

And for the SVD model, we obtained the following hyperparameters: {'n_factors': 150, 'n_epochs': 30, 'lr_all': 0.01, 'reg_all': 0.1} with an RMSE of 0.91.

3 Week 8

Although RMSE is a metric generally used to predict the accuracy of predictions, it still has some limitations.

One of the limitations of this metric is the fact that it does not take into account the rank of the recommended item, so a movie, in this case, that the user might like might have a lower RMSE score.

Besides this, another limitation of this metric is that it assumes that all users give ratings in the same way. But in reality, each user has a unique way of giving ratings. Thus, the accuracy of the metric may be affected.

After calculating Hit rate averaged across users, Precision@k, averaged across users, Mean Average Precision (MAP@k), Mean Reciprocal Rank (MRR@k), and Coverage, I obtained the following results:

Average Hit rate (User-based): 0.07
Average Hit rate (SVD): 0.37
Average Hit rate (TopPop): 0.63

Precision@5 (User-based): 0.01351
Precision@5 (SVD): 0.11808
Precision@5 (TopPop): 0.19259

Average MAP@5 (User-based): 0.00424
Average MAP@5 (SVD): 0.01244
Average MAP@5 (TopPop): 0.10919

Average MRR@5 (User-based): 0.02121
Average MRR@5 (SVD): 0.05320
Average MRR@5 (Top-Popularity): 0.38250

Coverage (User-based): 13.83%
Coverage (SVD): 6.17%
Coverage: 0.39%

All these metrics have both advantages and disadvantages.

The hit rate has the advantage of being easy to calculate and comprehend. However, it does not consider the rank of the recommended item.

Precision@k takes into account the ranking of the recommended items and gives more detailed information about the recommendation. But it doesn't consider the user's personal opinion about the relevance of the recommendation. And sometimes it might punish the system for suggesting relevant items that don't perfectly match the user's taste.

Mean Average Precision (MAP@k) takes into account both the relevance and position of recommended items so it gives more detailed feedback on the quality of the recommendations. But it takes more time to calculate and may not be suitable for big datasets.

Mean Reciprocal Rank (MRR@k) measures how quickly a relevant item appears in the recommendation list, giving a higher score to systems that suggest relevant items at higher ranks. However, it may not take into account the variety of recommended items and may not reward systems that suggest multiple relevant items at lower ranks.

Coverage shows the percentage of items that are recommended to at least one user. It helps to evaluate the diversity of the recommended items. But it may not work well for systems that suggest personalized items to each user.

Comparing these metrics for the 3 systems, User-based, SVD and TopPop, we can draw the following conclusions:

User-based has a low hit rate, precision, MAP, and MRR but has a high coverage.

TopPop has the highest hit rate, precision, MAP, and MRR and a high coverage.

SVD has a higher hit rate, precision, MAP, and MRR than user-based model, but still lower than TopPop , and it also has a low coverage.

Among all these 3 systems, the one with the best overall performance is TopPop. One reason TopPop works so well is that it is a very simple and easy to implement solution that does not involve complex calculations or algorithms. It simply recommends the most popular articles that are likely to be liked by many users, leading to high hit rates and accuracy.

Each of these 3 models comes with advantages and disadvantages.

User-based collaborative filter is a quick and easy implementation without the need for initial data modeling and also it can handle new users and items without the need for model retraining. However, it also has limitations, such as when it encounters the "cold start" problem where new users and items have insufficient data for accurate recommendations.

The SVD has the advantages that it can manage the problem of the data sparsity better than the user-based model, and it can make accurate recommendations for new users and items. On the other hand, this model may suffer from overfitting and requires more time-consuming data modeling to convert the user-item matrix into a lower-dimensional feature space.

TopPop system is a simple and easy to use algorithm, it does not require data modeling and can be used in comparisoning the recommendation models. On the other hand, it does not take into account the preferences of each user, but only recommends what is the most popular, thus suffering from popularity bias.

4 Week 9

.