COMP47580

Recommender Systems & Collective Intelligence

Recommender Systems Report

<Weijing Li>

<19204246>

Declaration of Authorship

I declare that all material in this assessment is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

Signed . .....Li Weijing..... Date .....2020/4/17.....

1 Non-personalized Model

1.1 Performance criteria

Figure 1 presents different performance criteria like coverage, recommendation coverage, item-space coverage and recommendation popularity versus four similarity metric including Genre, Genome, Rating and IncCondifence. In this section, I will analyze the value of evaluation criteria one by one.

The coverage is given by the percentage of target movies for which at least one recommendation can be made. In this case, only IncConfidence doesn't reach 1.00, because it has an assigned rating threshold which set to be 4 to justify whether the movie is liked by a user. If all of rating is smaller than 4, the target movie won't have recommendation.

The recommendation coverage is given by the percentage of movies in the dataset which appear at least once in the top-k recommendations made over all target movies. The genre metric uses overlap coefficient as follows to calculate the similarity, which means it focus on the intersection of the tags (e.g. [Adventure, Fantasy, Children]) between target movies. Some movies may have some tags that exist once on all movies for there is 1128 tags, so genre has the lowest recommendation coverage. Because of the threshold, the IncConfidence has a low value in this measurement.

$$\text{overlap}(\text{A}, \text{B}) = \frac{|A \cap B|}{\min(|A|, |B|)}$$

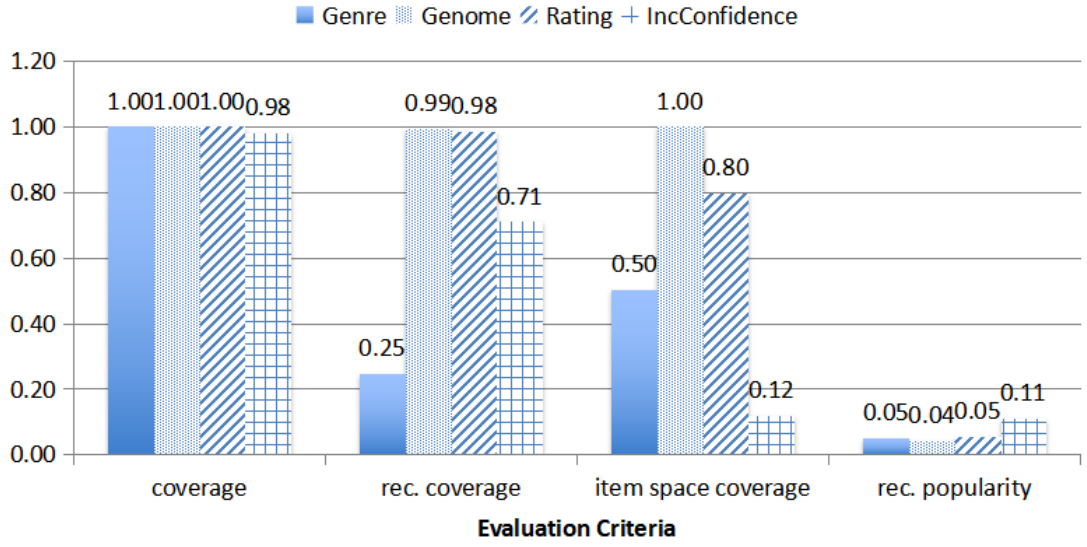

Figure 1. Performance criteria versus similarity metric

The item-space coverage is given by the average percentage of movies in the system which are capable of being recommended over all target movies. The IncConfidence has the lowest value, because for each target movie, it applies a threshold 4 in the rating with a maximum value of 5, so very few of the movies will be recommended each time. In some cases, the intersection of overlap coefficient may be an empty set, so genre has a low value. The Rating metric use cosine of the angle based on descriptions of the movies,

a small number of movies may vertical to each other, so the item-space coverage equals to 0.8. The Genome Metric calculates the similarity between movies using weighted Jaccard as follow, so all of the movies will exists in recommendation ranked by the output of weighted Jaccard for each target movie, so the value equals to 1.0.

$$\text{Jaccard(A, B)} = \frac{\sum_i \min(A(t_i), B(t_i))}{\sum_i \max(A(t_i), B(t_i))}$$

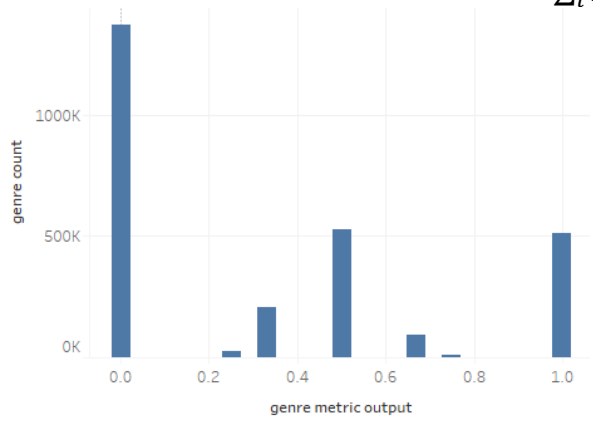

Figure 2. Genre metric output

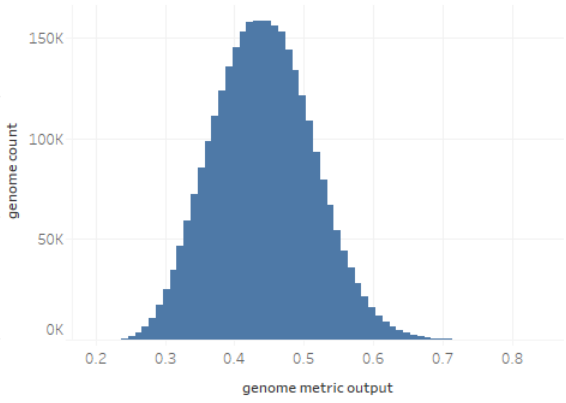

Figure 3. Genome metric output

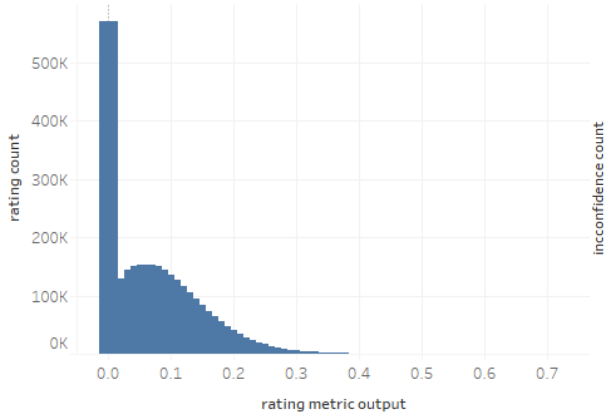

Figure 4. Rating metric output

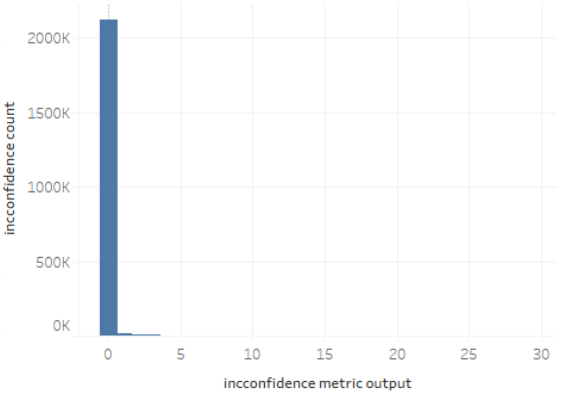

Figure 5. IncConfidence metric output

To verify the theoretical analysis, I draw four figures to present the distribution of the number in the similarity map of four different metrics. IncConfidence metric has 2,432,181 of outputs less than 1, while Rating and Genre has about 569,622 and 1,371,168 zeros separately, and IncConfidence doesn't have zeros. It's obviously that the output of similarity matric verifies the analysis, and meets the item-space coverage shown in Fig.1.

The recommendation popularity gives by the average of percentage of users in the system which have rated the movie over top-k recommended movie for each target movie. Since all recommendation methods is based on item-item similarity, genre and genome only based on tags and relevance of it, while rating and IncConfidence based on ratings from users.

1.2 Recommendation relevance

Figure 6 shows the recommendation relevance versus similarity metric. The recommendation relevance values estimate whether the recommendations satisfy user(s) preferences.

As can be seen from Figure, IncConfidence has the highest relevance value while the Genre has the lowest. The value is given by the average over all target movies, for each given target movie, the relevance value of top-3 recommendations made is calculated by taking the average of the mean rating over each recommended movie.

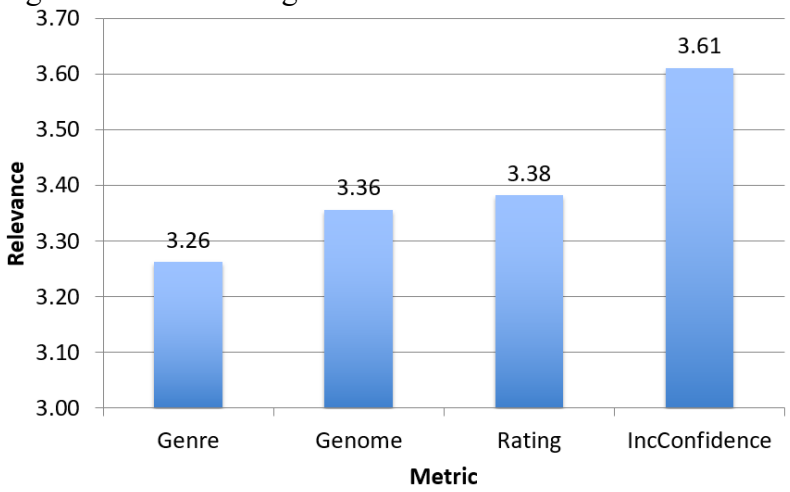

Figure 6. Recommendation relevance versus similarity metric

As analyzed above, Rating and IncConfidence based on rating of movies, Genre and Genome only depends on tags and does nothing with rating, so the relevance of genre and genome is lower than the other two. Genome aimed to ranking the recommended movies, while genre focuses on the number of same tags between movies, so Genome has better relevance value than Genre. Rating metric based on the cosine of the angle between ratings pattern, while the IncConfidence uses product association approach which set a threshold to count the number of used who rated the movie larger than 4, and use it to calculate the similarity, so it's obviously that IncConfidence can have a larger relevance value.

2 User-based collaborative filtering model

2.1 Different neighbour size

In this section, two figures perform the coverage and RMSE versus different neighbourhood size of four different predictors, including non-personalised approach, simple average approach, weighted average approach and deviation from user-mean.

Figure 7, starting from k=10, the Root Mean Squared Error of k nearest neighbour approach with different predictors decreased obviously, except non-personalised predictor, because it only considered movie-movie similarity, the number of neighbourhoods doesn't affect the accuracy of it.

Speaking of the trend of three user-based predictors, picking a small k results in poor accuracy, for there exists some users with few/no close neighbours. After the k becomes larger, the RMSE of user-based predictors decreases. The DFUM achieves the best performance while both SAP and WAP performs worse. Since Simple Average Approach only focus on average of neighbour's rating for target item, while Weighted Average Approach take the similarity between users into consideration, so the WAP is slightly better than SAP. Deviation from user-mean, which achieves best accuracy, rakes both user-user similarity and bias in user ratings into account.

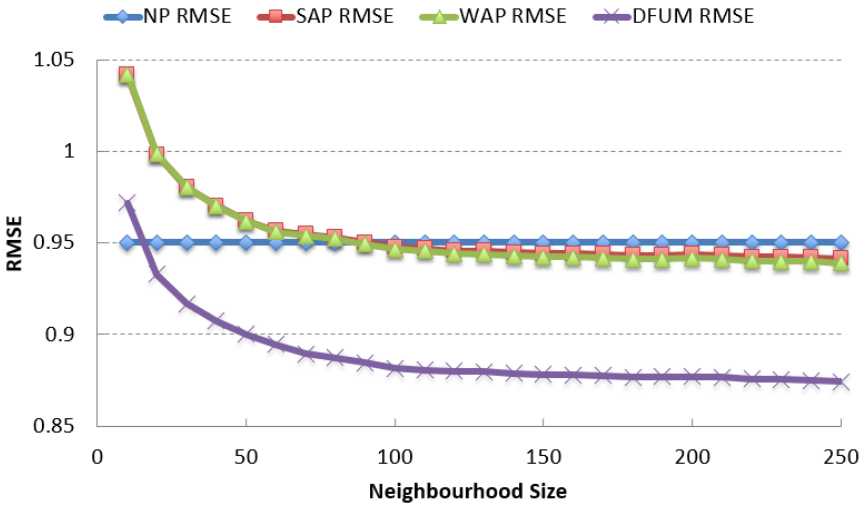

Figure 7. RMSE versus neighbourhood size

Speaking of Figure 8, coverage presents the percentage of users for whom predictions/ recommendations can be made. If k is small, in certain circumstances, all users in the neighbourhood set haven't rated the target movie yet, so the recommendations cannot be made. As analysed before, the NP predictor won't be affected by the neighbourhoods size. Besides, since the cosine similarity metric has been chosen to make the prediction, the coverage of NP is always 100%. For the other three predictors, we can see clearly that with the increase of the number of neighbourhood size, the coverage increase rapidly and achieves 100% simultaneously, because no matter which predictor is chosen, for target items, the similarity metric is the same.

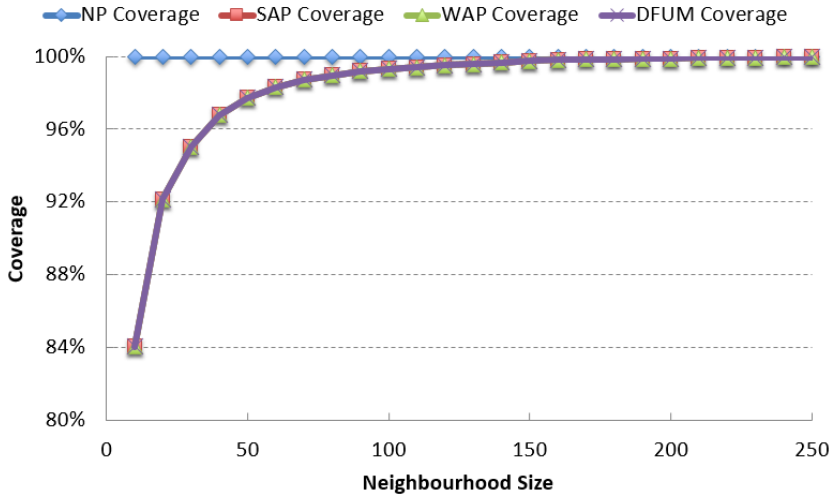

Figure 8. Coverage versus neighbourhood size

2.2 Different threshold

This figure shows the RMSE and coverage versus threshold neighbour approach with DFUM predictor and cosine similarity. With the increase of threshold, the number of neighbours will decrease, some of the neighbours that have small similarity values will be filtered.

Generally, the RMSE value will fall first and then rise, however the RMSE value in the figure is unexpected. It makes sense that coverage will decrease rapidly because after being filtered, fewer users can get their recommendations, but the RMSE only decreases slightly at the first four points. This may due to the particularity of the data, leading to the overfitting of the model starts from threshold 0.15. The rapidly decrease of RMSE at the threshold 0.8, may be accidental, because only 2 users can have their recommendations at this point.

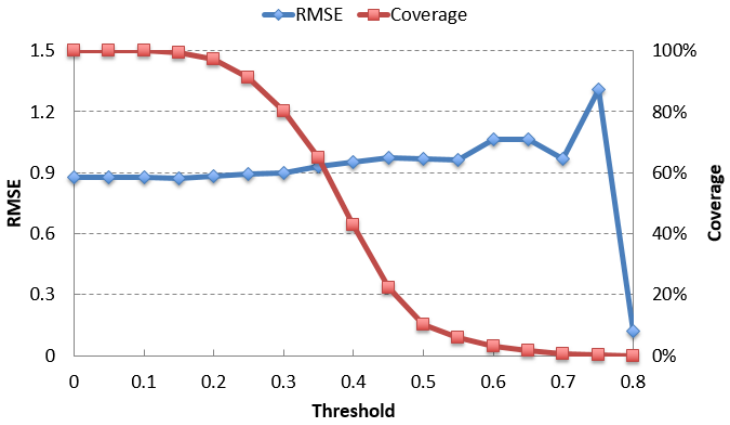

Figure 9. RMSE and coverage versus threshold

2.3 Different similarity metric

Figure 10 shows the RMSE and coverage versus different similarity metric with 200-nearest neighbour approach combined with deviation from user-mean predictor. Pearson and Cosine metric get coverage

It is clearly that Pearson and MSD achieves about 6% higher accuracy than Cosine and Pearson, but speaking of coverage, the Pearson Sig and cosine metric get more than 5% higher than Pearson and MSD.

It's because Pearson and MSD only focus on the intersection of profiles, weights that are calculated over small numbers of co-rated items may provide an unreliable measure of the similarity between users. Pearson Sig improves the Pearson metric, modifying similarity weights based on the number of co-rated items between users. Besides, cosine similarity ensures than users who have rated many items are not a priori more similar to other users, so it also achieves a good accuracy.

On the other hand, Pearson and MSD doesn't take the weight of numbers of co-rated items into considered, also cause the lower value of coverage.

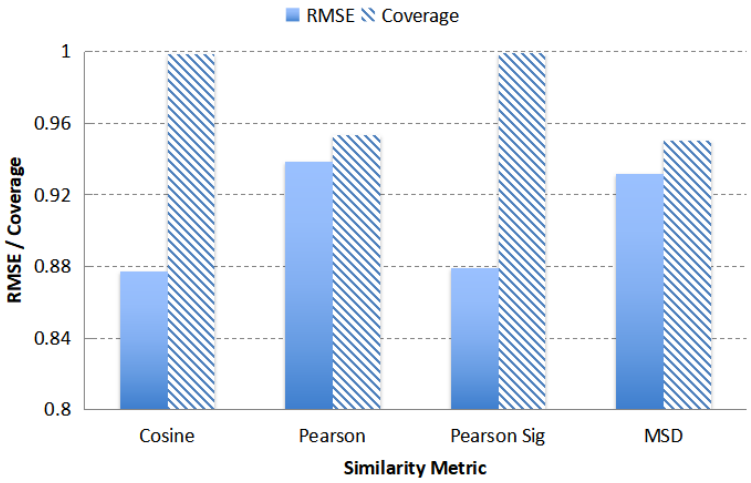

Figure 10. RMSE and coverage versus similarity metric

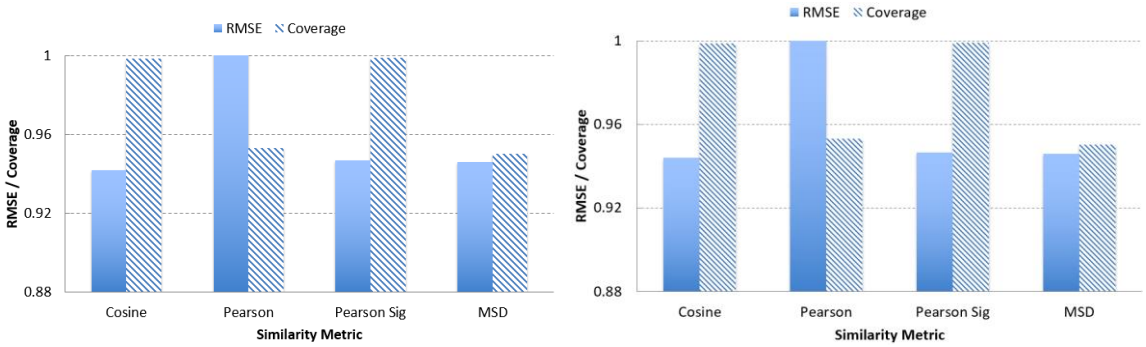

Figure 11. RMSE and coverage versus similarity metric with WAP/SAP predictor

In order to understand the impact on the similarity matrix on the coverage and RMSE results, so I combine the similarity metric with different predictors. It is clearly that no matter what personalized predictor is chosen, the coverage won't change, it verify that with the same neighbour size, the coverage values depends on the similarity metric.

Reference

[1] Zisopoulos, Harry & Karagiannidis, Savvas & Demirtsoglou, Georgios & Antaris, Stefanos. (2008). Content-Based Recommendation Systems.