추천 시스템의 협업 필터링: 아이디어와 평가

김주성
코너스톤 서울아카데미

Collaborative Filtering in Recommendation Systems: Idea and Evaluation

Joosung (James) Kim
Cornerstone Collegiate Academy of Seoul

요약

Collaborative filtering has been used frequently as a recommendation system. To reduce the errors on predicting the ratings that may be given by the user, we propose a new aggregation method to do so. We used a real-world dataset MovieLens to compare our proposed method from previously existing methods, and accordingly to the results, ours was more accurate.

1. Introduction

The definition of recommendation systems, in general, is one that provides a user with an item that he or she would probably like, an act that is based on the user's past evaluation, purchase, browsing history, or all [1][7]. Figure 1 is a small example of an offline recommendation.

Figure 1: offline recommendation

This technology is implemented in many different websites such as the following: Amazon (illustrated in Figure 2), where the user's daily searching and purchasing history is analyzed and will contribute in recommending more products in the future; and Netflix (illustrated in Figure 3), in this case, the user's ratings on different movies are analyzed to recommend new ones.

There are 3 different classifications of recommendation systems: content-based, collaborative filtering, and trust-based [1][4][5]. A content-based approach is a way of recommending items that have the same characteristics of items in the user's history. If a target user watches a significant number of horror movies, there will be more recommendations of horror movies based on his or her history of watching horror movies. A trust-based approach is one that is based on the relationship between the user and his or her trustable users with either a direct or indirect trust relationship with the target user. More specifically, if there is a group of people a target user considers as "trustworthy" in means of similarities in preference, the items the group was fond of will also be recommended to the target user. The approach we will focus on is collaborative filtering (CF) [8], in which the user is recommended with the items that are rated high by his or her neighbors who are other users with similar preferences to that of the user.

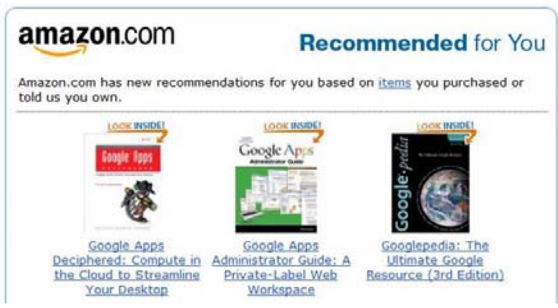

Figure 2: Recommendation in Amazon.com [2][8]

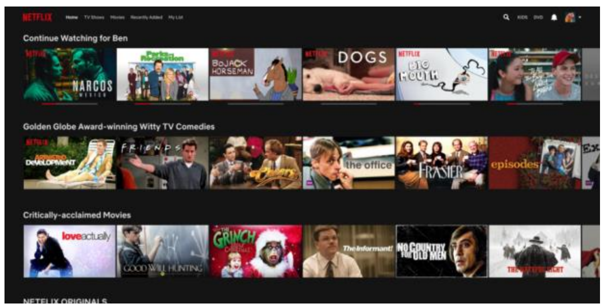

Figure 3: Recommendation in Netflix.com [3]

The CF technique has been one of the most utilized approaches in recommendation systems through the process of rating predictions and top-N recommendations. The goal of rating predictions is to predict accurately the original ratings on items of a user before recommendation. The goal of top-N recommendations, on the other hand, is to predict relative preferences (i.e., not actual ratings) on items of a user for selecting the top-N most preferred items. In this paper, we will be focusing on rating predictions instead of top-N recommendations.

A CF technique has three main steps of its progress: finding a group of people with similar tastes (step 1), estimating the ratings using an aggregation method (step 2), and recommending items deemed high (step 3). In this paper, we have two contributions: we propose a new method for the aggregation step (step 2) to reduce the errors in rating predictions; also, we will perform an evaluation to compare existing and proposed methods for rating predictions.

The paper is organized as follows: Section 2 introduces collaborative filtering and proposes our aggregation. Section 3 presents the results of accuracy evaluation. Section 4 concludes our work.

2. Collaborative Filtering

In this section, we explain collaborative filtering and present our idea to enhance the recommendation accuracy.

As mentioned before, collaborative filtering is a method that exploits neighbors (a group of users whose preferences are similar to that of target user c) of target user *c*. It is performed in three steps:

1. Find neighbors.
2. Estimate $r_{c,s}$, the rating of item s for target user c, based on the ratings given to item s by c's neighbors $\hat{C}$.
3. Recommend the items whose ratings are estimated high to the target user.

In (step 1), it is necessary to calculate the similarity value of the two users, target and a possible neighbor. For this, we need a similarity measure. We have two options: Pearson Correlation Coefficient (PCC) and Cosine similarity. PCC and Cosine similarity are defined as the following [1][6]:

$$sim(x,y) = \frac{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)(r_{y,s} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)^2 \sum_{s \in S_{xy}} (r_{y,s} - \bar{r}_y)^2}}$$

$$sim(x,y) = \frac{\sum_{s \in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2}\sqrt{\sum_{s \in S_{xy}} r_{y,s}^2}}$$

$S_{x,y}$ is the set of all items rated by both the target user and his or her neighbor. $r_{x,s}$ and $r_{y,s}$ are the ratings given by each user to the specific item, while $\check{r}$ represents the average rating given by the user on different items.

In PCC, the coefficient represents the proximity of the coordinates from the linear y = x graph, when the y-axis stands for the target user's ratings and the x-axis represents the active user's ratings on different items. In Cosine similarity, if x items are rated by both a target user and a candidate neighbor, x number of axes are made. From this, there will be two points representing each user's ratings on the items. Then, when two vectors are made by linking each vector from the origin, the cosine of the angle becomes the similarity.

In (step 2), using different forms is also possible in finding relations with the ratings given by the neighbors. Three popular methods of aggregation used in the past are the following [1]:

M1: $r_{c,s} = \frac{1}{N} \sum_{c' \in \hat{C}} r_{c',s}$

M2: $r_{c,s} = \left(\sum_{c' \in \hat{C}} sim(c,c') * r_{c',s} \right) \Big/ \sum_{c' \in \hat{C}} sim(c,c')$

M3: $r_{c,s} = \bar{r}_c + \sum_{c' \in \hat{C}} sim(c,c') * (r_{c',s} - \bar{r}_{c'}) \Big/ \sum_{c' \in \hat{C}} sim(c,c')$

M1 represents a regular, ordinary, and fundamental process of simply averaging out the ratings on the item of the neighbors. However, some ratings can have a greater significance than other ratings. In order to emphasize the significance, a new way of weighted averaging is implemented with M2. On the other hand, in this case, there is no consideration on a significance of how much more or less the neighbor has given the ratings on average compared to others. In other words, different users rate accordingly to their distinct way of rating items. In order to include this factor, M3 was implemented, which includes the deviation from the average rating of the neighbor. It is used to overcome the difference in the average ratings given by individual neighbors.

In this work, we propose a new method, M4, for the aggregation step. It is formulated in the following:

M4: $r_{c,s} = \bar{r}_c + \sigma_c * \left(\sum_{c' \in \hat{C}} sim(c,c') * \frac{(r_{c',s} - \bar{r}_{c'})}{\sigma_{c'}} \right) \Big/ \sum_{c' \in \hat{C}} sim(c,c')$

This equation includes the standard deviation of the target user and the neighbors. It considers how much the rating of the neighbor on an item deviate from his or her usual pattern of ratings in the unit of his or her standard deviation. We create M4 to overcome the differences in variance of users. Someone may have a wide distribution of ratings, while someone else may have narrow distribution of ratings.

3. Evaluation

In this section, we show the results of our conducted experiment by comparing different outcomes. In this experiment, we used a real-world dataset, MovieLens, tested in the means of evaluating recommendation systems. To conduct our experiments, we used 80% of the total ratings as a training set and the other 20% as a testing set. In order to evaluate the effectiveness, we utilized the mean average error (MAE) and the root mean squared error (RMSE) [1][7] as the accuracy metrics, which are defined as below:

$$MAE = \frac{\sum_{(u,i)\in E} |\hat{R}_{u,i} - R_{u,i}|}{|E|}$$

$$RMSE = \sqrt{\frac{\sum_{(u,i)\in E} \left(\hat{R}_{u,i} - R_{u,i}\right)^2}{|E|}}$$

The variables *u* stands for the user, while *i* stands for the item. E is the ratings in the dataset MovieLens, and $R^{\wedge}{}_{u,i}$ is for the predicted ratings on user u on item i.

We have three factors (aggregation methods, number of neighbors, similarity measures) to be tested. Table 1 shows the values and a pivot (in boldface) for each factor.

Factors	Options
Aggregation	M1, M2, **M3**, M4
Similarity	**PCC**, Cosine
k- value	5, 10, **30**, 50

Table 1. Factors and their values

For aggregation methods, we have M1, M2, M3, and M4. For the number of neighbors, we used the variable k by 5, 10, 30, and 50. For similarity measures, we used PCC and Cosine similarity. The results obtained from the experiments are recorded and analyzed via the graphs shown below. For our experiments, we borrowed the code in [9] made available to the public.

Figures 4 and 5 compare aggregation methods in rating prediction in terms of RMSE and MAE. We fixed the value of k as 30 and the similarity as PCC. Our proposed aggregation method, M4, yielded the least amount of errors, thus being the most accurate of all aggregation methods. The order in terms of decreasing accuracy is as follows: M4, M3, M1, and M2. The accuracy of M2 being lower than that of M1 was quite interesting, given that we expected the opposite result.

Figure 6 is a representation of the difference in errors between PCC and Cosine. It shows that PCC shows that it is slightly more accurate compared to the Cosine similarity.

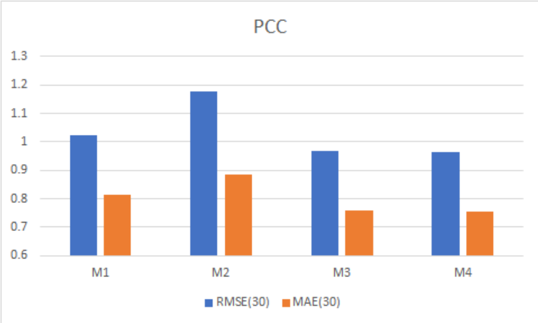

Figure 4. The error with different methods in PCC/k=30

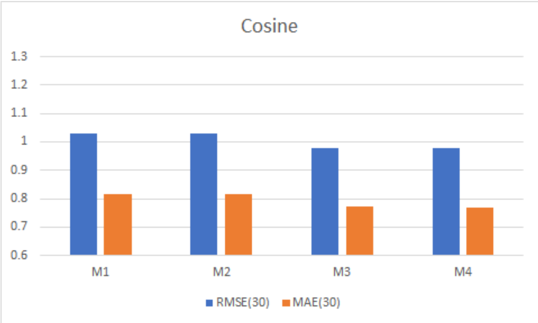

Figure 5. The error with different methods in Cosine/k=30

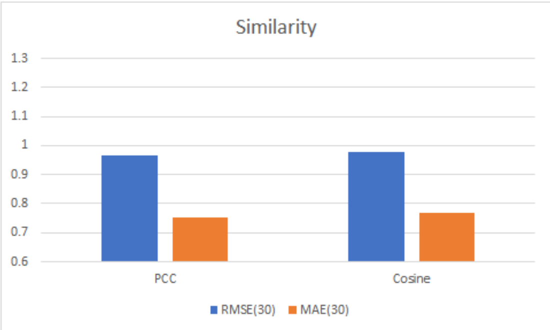

Figure 6. The error with different two similarities

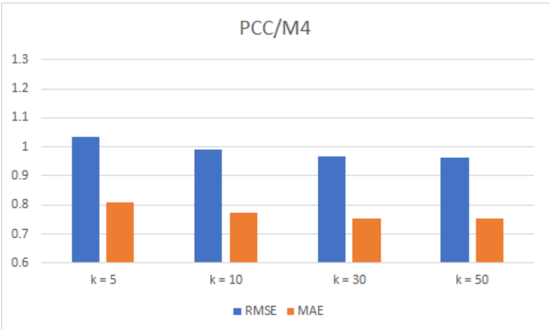

Figure 7. The error with different k values

Figure 7 shows the error values of different k values. It concludes that k = 50 is the most accurate of all the values of k. However, if the value of k increases beyond 50, the accuracy will decrease. M4 overall was the method with the best accuracy in terms of both RMSE and MAE.

4. Conclusions

Collaborative filtering is a version of recommendation system that is utilized in various businesses such as Netflix and Amazon. It uses three steps in doing so: finding neighbors, aggregation, and recommendation. In this paper, we propose a new aggregation method, M4, and verify its effectiveness in reducing the errors in terms of RMSE and MAE on a real-life dataset MovieLens. We concluded that M4 had the least amount of errors, thus being the best out of the 4 aggregation methods of collaborative filtering in effectiveness.

5. Reference

[1] Gediminas Adomavicius, Alexander Tuzhilin: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering. 17(6): 734-749 (2005)

[2] Amazon.com

[3] Netflix.com

[4] Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, John Riedl: Item-Based Collaborative Filtering Recommendation Algorithms. WWW 2001: 285-295

[5] Mukund Deshpande, George Karypis: Item-based top-N recommendation algorithms. ACM Transactions on Information Systems. 22(1): 143-177 (2004)

[6] Ricardo A. Baeza-Yates, Berthier A. Ribeiro-Neto: Modern Information Retrieval. ACM Press / Addison-Wesley 1999, ISBN 0-201-39829-X

[7] Charu C. Aggarwal: Recommender Systems - The Textbook. Springer 2016, ISBN 978-3-319-29657-9, pp. 1-498

[8] Greg Linden, Brent Smith, Jeremy York: Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Internet Computing 7(1): 76-80 (2003)

[9] https://sites.google.com/view/usercf