

Research on collaborative filtering recommendation algorithm based on temporal context

Jun Li*, Ge Yu[#]

School of Information Science and Technology, Hangzhou Normal University, Hangzhou, China

ABSTRACT

Collaborative filtering-based algorithms are widely used to make recommendations without analyzing the contents. Time effect can be seen everywhere in our daily life. User interests will change over time, so we use the time-decay function to integrate the user-item rating matrix and adjust it by different time-decay factors to optimize the model. And we conducted experiments using the improved algorithm on the movie evaluation dataset movielens-1m. The results show that the algorithm is able to improve the accuracy and coverage of recommendations under specific time factors, and also can partly improve the recommendation efficiency.

Keywords: Collaborative filtering, time effect, movie recommendation, time decay, time factor

1. INTRODUCTION

In recent years, with the rapid development of the Internet, more and more data are stored in the network. The rapid development of web applications, especially mobile applications, enables people to easily browse a large number of online information resources. Therefore, how to recommend resources (e.g., commodities, movies, books, etc.) that meet users' needs from the vast amount of information resources has become one of the current concerns of researchers¹. Recommendation systems play a very effective role in filtering and screening information, and can effectively alleviate the information overload problem. Among the current research on recommendation, collaborative filtering is a basic recommendation algorithm and has been widely used in academia and industry. Despite this, there are still inevitable problems such as data sparsity, cold start, poor real-time performance, and outdated information². To address the outdated information problem, the traditional collaborative filtering algorithm makes recommendation based on users' historical rating data and does not fully consider the influence of the time factor³. In this paper, we take the time effect of human interest as the starting point, and construct a time decay function by comparing different time decay factors to combine with collaborative filtering for more time-efficient recommendation.

2. RELATED WORK

2.1 Collaborative filtering

Collaborative filtering is one successful personalized recommendation technology, and is widely used in many fields⁴. The recommendation process that collaborates with everyone's feedback, reviews and opinions to filter the massive amount of information together and filter out the information that may be of interest to the target user⁵.

- User-based Collaborative Filtering (User-CF-Base)
- Item-based Collaborative Filtering (Item-CF-Base)

We use UserCF as the basis for our hybrid algorithm.

2.2 User similarity calculation

Similarity calculation is a key step that affects the performance of the recommendation algorithm. There are many methods to calculate similarity, among which the cosine similarity method is the most classical and especially suitable for data sparse situations, so we choose the cosine similarity algorithm to calculate the user similarity⁶. For users u and v the cosine similarity is calculated as shown in equation (1):

*2020112011023@stu.hznu.edu.cn; #yuge@hznu.edu.cn

$$w_{uv} = \frac{|N(u) \cap N(v)|}{\sqrt{|N(u)||N(v)|}} \quad (1)$$

where $N(u)$ denotes the set of items which user u has rated, $N(v)$ denotes the set of items which user v has rated, and w_{uv} denotes the cosine similarity between user u and user v .

The existing collaborative filtering-based recommendation algorithms have a problem: they always only focus on the similarity between users or resources, but ignore the dynamic changes of users' interests. But in real life, users' needs for resources change over time⁷, and traditional collaborative filtering algorithms only use the user-resource access matrix for recommendation calculation, without considering the specific time of users' access to resources. When the user's interest changes, the existing recommendation system cannot detect it in time, so the recommended resources may be largely deviated from the user's demand.

In this paper, we mix time factor and user collaborative filtering (Time-UserCF) to perform analytical comparison experiments.

2.3 Time effect

Time effect can be seen everywhere in our daily life. For example, dressing styles vary with age; dressing changes with the seasons, and so do fruits.

2.4 Preference migration

Due to users' own reasons, their interests change over time. The user's preferences directly affect the recommended result set. Here we select the trend of the average rating score change of movies corresponding to user with uid=1086360 from the Netflix dataset between September 2002 and December 2005 as shown in Figure 1.

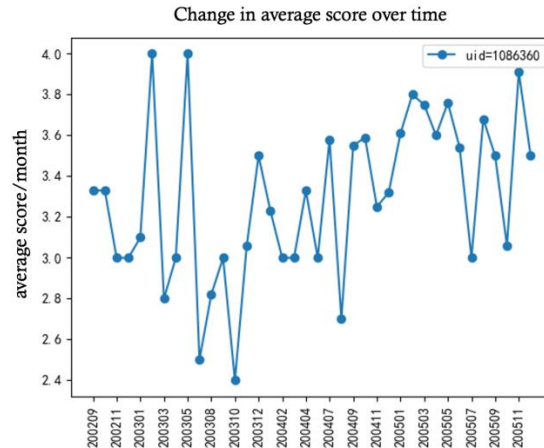


Figure 1. Trend of monthly average rating score change for users with uid=1086360.

From Figure 1, we can see that the trend of user's rating score change is different at each time point. Therefore, the corresponding time function should be introduced to simulate the change of user's interest when rating the items.

3. COLLABORATIVE FILTERING ALGORITHM WITH TIME DECAY FUNCTION

In this paper, we focus on the hybrid algorithm of UserCF. When using UserCF to recommend items for users, we first find the user set with similar interests to the target user, and then make recommendations for the user based on the behavior of these users⁸. The formula of user similarity calculation is shown in equation (2):

$$w_{uv} = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|} \quad (2)$$

where $N(u)$ denotes the set of items for which user u has generated behavior, $N(v)$ denotes the set of items for which user v has generated behavior, $|N(u) \cap N(v)|$ denotes the number of items for which user u and user v have intersection, and $|N(u) \cup N(v)|$ denotes the number of union items for which user u and user v have generated behavior.

When calculating user similarity, a penalty term should be added for popular items, and here the user similarity formula can be modified as equation (3):

$$w_{uv} = \frac{\sum_{i \in N(u) \cap N(v)} \frac{1}{lg(1+N(i))}}{|N(u) \cup N(v)|} \quad (3)$$

where $N(i)$ denotes the number of all users who have generated an action on item i .

Because the behavior in the most recent period best represents the user's current interest⁹, a temporal decay function is added to calculate the similarity between the two users as equation (4):

$$w_{uv} = \frac{\sum_{i \in N(u) \cap N(v)} \frac{1}{lg(1+N(i))} f(|t_{ui} - t_{vi}|)}{|N(u) \cup N(v)|} \quad (4)$$

where $f(|t_{ui} - t_{vi}|)$ is the time decay function¹⁰, which has the form given in equation (5):

$$f(|t_{ui} - t_{vi}|) = \frac{1}{1 + \alpha |t_{ui} - t_{vi}|} \quad (5)$$

where α is the time decay factor, t_{ui} denotes the time when user u has behavior on item i , and t_{vi} denotes the time when user v has behavior on item i .

Since the current rating receives a relatively large influence from the recent ratings of the similar user set. Therefore, the time decay function $f(|t_0 - t_{vi}|)$ is also added to calculate the user's rating of the item, so the final interest degree of user u in item i is obtained in equation (6):

$$r_{ui} = \frac{\sum_{i \in N(u) \cap N(v)} \frac{1}{lg(1+N(i))} f(|t_{ui} - t_{vi}|)}{|N(u) \cup N(v)|} \times r_{vi} \times f(|t_0 - t_{vi}|) \quad (6)$$

where t_0 denotes the current time and t_{vi} denotes the time when user v generates an action on item i .

4. EXPERIMENT

4.1 Experimental environment

Using a desktop computer with 32GB of RAM and Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz; operating system is windows 10, using PyCharm development tool with python version 3.8 .

4.2 Dataset

Dataset is Movie Lens-1M, which is a movie rating dataset collected by the Group Lens research team from the Movie Lens system at different time periods. The data is authentic and valid, including movie attribute information, user demographic information, and movie rating score which is the most critical information for our experiment.

4.3 Experimental design

This experiment is modified on the traditional UserCF algorithm. The data are split into training and test sets by calling `train_test_split` in `sklearn`, where `test_size` is 0.1 and `random_state` is 40. The time decay function $f(|t_{ui} - t_{vi}|)$ is added to calculate the user similarity matrix, and by modifying the time decay factor α and the number of nearest neighbor users k for experimental comparison, and we also adding the time decay function $f(|t_0 - t_{vi}|)$ during recommending. The time decay factor α for training is 0.5 and 5, and the time decay factor β for recommendation is 0.8 and 10, which are divided into two groups, (0.5, 0.8) and (5, 10). These two groups use the number of {5, 15, 25, 35, 45} nearest-neighbor users to compare the experiment with the traditional UserCF.

Due to the large number of full-scale test users, only the accuracy, recall and coverage of 10 random users are tested in conducting the evaluation metrics, and then the final evaluation metrics are obtained by repeating the experiment 5 times and calculating the arithmetic mean.

5. RESULTS

The experimental results when the number of nearest neighbor users $k = \{5, 15, 25, 35, 45\}$ are shown in Figures 2-4.

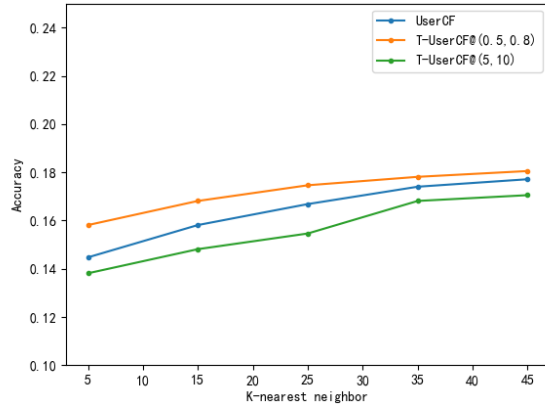


Figure 2. Figure with accuracy results.

The figure above shows that the accuracy increases slowly with k , and adding the time decay function in the algorithm, the accuracy is improved under the time decay factor $(\alpha, \beta) = (0.5, 0.8)$, but in the case of time decay factor $(\alpha, \beta) = (5, 10)$, the accuracy rate has decreased.

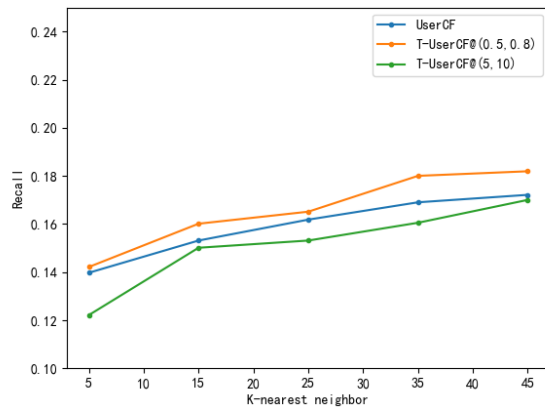


Figure 3. Figure with recall results.

The figure above shows that the recall increases slowly with k , and adding the time decay function in the algorithm, the recall is improved under the time decay factor $(\alpha, \beta) = (0.5, 0.8)$, but in the case of time decay factor $(\alpha, \beta) = (5, 10)$, the recall rate has decreased.

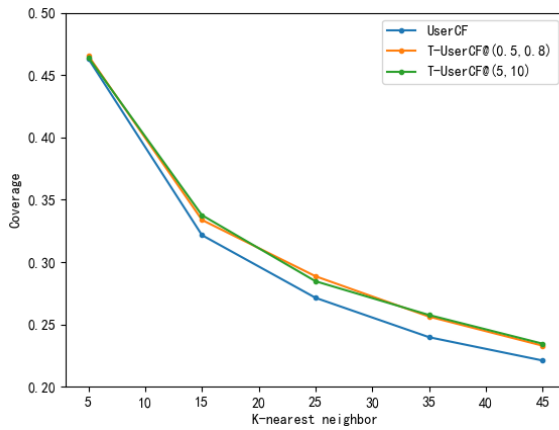


Figure 4. Figure with coverage results.

The figure above shows that the coverage decreases slowly with k , and adding the time decay function in the algorithm, the coverage is improved to some extent. The time decay factor has little effect on coverage.

6. CONCLUSION AND FUTURE

The experiments show that by modifying the time decay factor and adding the time decay function in the algorithm, the accuracy and recall are improved under the first time decay factor, and the recommendation efficiency has also been improved. However, in the case of the second time decay factor, the accuracy and recall rate are lower than the traditional algorithm, the coverage is both improved compared with the traditional algorithms, and these two time-decay factors have little effect on the coverage.

In the future, we will use deep learning methods to improve the collaborative filtering algorithm on temporal context, considering the change of user's interest to improve the accuracy and coverage, and enhance the system efficiency.

REFERENCE

- [1] Yu, M., He, W. T., Zhou, X. C., Cui, M. T., Wu, K. Q. and Zhou, W. J., "Overview of recommendation system," *Computer Application*, 1-16 (2021).
- [2] Lan, Y. and Cao, F. F., "Research on time weighted collaborative filtering algorithm for film recommendation," *Computer Science*, 44(04), 295-301+322 (2017).
- [3] Zhang, Z. F., Tang, M. and Li, Q. X., "Research on personalized film recommendation algorithm based on time context," *Journal of Hebei Software Vocational and Technical College*, 22(04), 5-8 (2020).
- [4] Fang, P., "An electronic commerce collaborative filtering recommendation algorithm based on user context," *Applied Mechanics and Materials*, 3634, 687-691 (2014).
- [5] Xing, C. X., Gao, F. R., Zhan, S. N. and Zhou, Z. Z., "Collaborative filtering recommendation algorithm to adapt to the change of user interest," *Computer Research and Development*, (02), 296-301 (2007).
- [6] Liang, S. Y., Peng, X. L., Qin, B., Lin, W. M. and Hu, Z. N., "Collaborative filtering book recommendation based on time context optimization," *Library Forum*, 41(03), 113-121 (2021).
- [7] Zhao, H. Y., Liu, Q. Y., Chen, Q. K. and Cao, J., "Matrix decomposition algorithm based on similarity decay with time," *Small Microcomputer System*, 37(07), 1474-1478 (2016).
- [8] Liu, Y., Wang, Y., Qi, G. T. and Bao, Z. Y., "Collaborative filtering top-N recommendation algorithm in time context," *Computer Technology and Development*, 27(07), 79-82 (2017).
- [9] Deng, Z. P. and Dai, Z. Q., "A catering intelligent recommendation model with improved temporal context correlation," *Software Guide*, 13(11), 152-153 (2014).
- [10] Xia, Z. Q. and Wang, Z. T., "Entropy driven collaborative filtering algorithm integrating time context," *Journal of Chongqing University of Science and Technology (Natural Science Edition)*, 21(03), 73-77 (2019).