**Time-sensitive collaborative filtering algorithm for Stable interest**

[College of Computer Science and Technology](https://www.baidu.com/link?url=6_Wjpm8Po16wMbB6r00ZEOMqx9gdC_xUNIo7PrEckDVCk2iltkPpTJkuffqt3nOV2m3Dg3uxIrINyWGrdbo5Bp4BuU3r_lQM6Mr0b4K8huBIac3cuf1tpGDsCuz9mD7Dqa_uvTPStURabxgmVlf3LyRNfGpnW5LHIEEodde9QCi&wd=&eqid=e618908600056d820000000458817ed8), Shandong University of Technology, Zibo 255049, China

.Collaborative Filtering Algorithm

**Abstract**：Aiming at the popularity bias phenomenon，introduce the prevalence of item into user interest modeling, and put forward an item popularity model based on user interest feature. Aiming at the problem that the traditional model does not take into account the stability of the user's interest and the difficulty of capturing the user's interest, the time-sensitive and the interest stability are introduced in the process of calculating the similarity of the user's interest. Introduce time-sensitive and stabilized interest similarity model. Finally, weighting those two kind of similarity model with weight factor，and a novel algorithm model is proposed - IPSTS Algorithm. Experiment show that, the result of the algorithm, Mean Absolute Difference (MAE) and root mean square error (RMSE) are significantly reduced than those of the Traditional Collaborative Filtering Algorithm and the long tail items can be excavated as well, which ease popular objects bias phenomenon.

**Key words:**time-sensitive; stability of interest; prevalence of item; collaborative filtering; popularity bias

**关键词** ：时间敏感；兴趣稳定性；物品流行度；协同过滤；流行偏置

**1 Introduction**

The Statistical Report on Internet Development in China shows Chinese netizen reached 667 million scale until June 2015[1]. With the rapid development of information technology, the amount of information also occurred explosive growth, information overload problem has become increasingly prominent. In the massive information, it is difficult for users to find what they interested part. Generally the technologies to solve this problem are search engine, information filtering technology, recommendation systems, at el. However, search engine usually used in the case of definite requirement situation given by user; dig out some partial user interested information. The recommendation system is to build a user interest modeling by analyze the user's historical behavior in data without explicit user requested input, and then make recommendations for users.

Collaborative Filtering Recommendation Algorithms are mainly divided into two categories: user based and item based. The idea of Item Based C​collaborative Filtering Algorithm is that, according to the user historical behavior, making data analysis and calculation, get a user's behavior preferences and make recommendations for users with the preferences. But the premise of this algorithm should that, users can not change the interest within a period of time. The idea of Item Based Collaborative Filtering Recommendation Algorithms is that, based on the behavior data of the user A, to calculate the neighbor users who have similar preferences with the user A, and then recommend the items - which are of interest to the neighbor user and the user A has not yet found. Therefore, no matter which CFRA, the core is the calculation of similarity.

Aiming at the dynamics problem in collaborative filtering technology, a recommendation algorithm, which adapt to user interest, is proposed in literature [3] - where the time weight and Data weight of resource similarity are weighted. literature [4] proposed a dynamic recommendation technique. As the mobile device flourishes，a SUCM model is proposed in [5], which learn fine-grained user preferences to make recommendations to users.

Generally, there are two traditional solutions to the problem of scalability in collaborative filtering technology. The first idea is based on clustering. In literature [6], it propose to set up a representative user for each cluster, and select the neighbor user set, by calculating and representing the similarity of the user. There is a methods based on deep learning to solve the problem of scalability in [7]. In [8], a hierarchical Bayesian network with cooperative depth learning is proposed to solve the problem of data sparseness.

The second idea is, solve the scalability problem by reducing Scoring Matrix dimensionality in literature [9]. In [10] it propose the use of principal component analysis techniques in statistics, to achieve the scale matrix dimensionality reduction.

For the problem of cold starting, the literature [11] proposed the use of proximity, influence and popularity of three aspects, to consider the influence of user rating on the similarity between users. In [12], a new heuristic similarity calculation model is proposed, which can alleviate the cold start problem and improve the similarity between users. In literatures [13], [14] and [15], solve the cold start problem with self-coding, improve the quality of top-N recommended. In [16], the social network is used to analyze the strong and weak relationships in the social community, and an EM algorithm is proposed to improve the recommendation quality. In [17], using depth learning technique to generate a learning function between the content and the user interaction, and the user preference is converted into a ranking list which is recommended to user.

On the issue of popularity bias, literature [18], it eases the problem by changing the popularity distribution of recommended result. In this essay, firstly, boxing all items popularity - "boxelization", and then according to the user's actual score to map the "box". After mapping, generate the formation of a three-dimensional vector, stand for object popularity based on the user interest feature - "vectorization". Finally, calculate the similarity between users.

**2 Method Descriptions**

Generally speaking, the higher popularity of the item, the user will be more interested in it, that is, user interest is related to the popularity of items.

In this paper, the popularity of items as a user interest characteristics to build the model, based on the popularity of items of interest similarity model of user interest. In this paper, considering items popularity as the user interest feature to build a model, propose the user interest similarity feature model based on items popularity. This article holds that the user's short-term interest is not necessarily the same, which needs to be determined according to the stability of the user's interest. If the user's interest is stable, then the user's interest will not change with the past of time. If the user's interest is unstable, the interest will be varied. This paper from the perspective of popularity and interest stability, proposes a time-sensitive collaborative filtering recommendation algorithm for stable user interest

**2.1 User Interest Feature Similarity Model Based on Item Popularity**

**Definition 1** (The popularity of Item i): The ratio of the number of reviews to the total number of items, to the total number of items. The formula is as follows:

 (1)

stands for the number of user who had reviewed item *I*, stand for total number of items.

**Algorithm Description:**

(1) Take 3 intervals of items popularity,, .

(2) Using formula:calculate all items popularity which had been reviewed by user.

(3) Boxing all the items popularity derived in step (2), and then, mapping in step (1).

Boxing pseudo code as following:

 1 





















Declare: ，，，，， are based on the experiment to obtain the threshold data

The Mapping pseudo code is:









































(4) Calculate the user interest feature similarity based on item popularity with User Feature Vector derived from step (3), using cosine similarity formula as following:

（2）

The above Model called Item\_pop\_sim Model (Item Popularity Similarity Model), IPS Model in brief。

**2.2 Time - Sensitive Similarity Model of Stable Interest**

In the actual application process, the user's interest is usually volatile, not only with the user on the item score value, but also with the popularity of items. The user interest constituted with the weight of the two factors. User interest is defined as follows:

**Definition 2** (The Interest of User  ) The Interest of User is the vector set  constituted with Vector of Interest which is The Interest Vector of User  for  Items.

**Definition 3** (The Interest Vector of User for item) The sum of the ratio of the Score of user  for item  to Full Score, and Popularity for this item. Formula as following:

 （3）

（4）

Stand for The Popularity for user  to item , stand for Score for user  to item ,  stand for the Full Score to the item.  is the Popularity for item  in formula (1).  stand for the Minimum Popularity for all items, while  is the Maximum Popularity. , are parameters derived from experimental results, and meet condition: . Calculate the Interest Similarity for users using Cosine Similarity as following:

 (5)

The model can effectively alleviate the problem of user scoring by introducing the popularity of items with weights.

**Definition 4**（Stability of Interest to User  ）The variance of score for user  to all item.

 （6）

 （7）

stand for the score to item ,  is total number of item reviewed by user , is the average value for all items.

The Variance used for measuring the stability of the user’s interest, that to say, the smaller the variance, the more stable the user’s interest.

Factors of interest to users: personal factor, time and environment. In reality, the user's interest is actually volatile, may be affected by their own factors, the surrounding environment, friends and family influence, interest will be subtle over time, long ago the interest may gradually fade or disappear. If the user's interest is relatively stable, this article that the user changes over time interest will not have much change. Therefore, from the factors that affect the user's interest, given the time-sensitive characterization.

**Definition 5** (time-sensitive)On the basis of the stability of the two users, the closer of review time of the user's score to the item is, the higher the similarity between the users, that is, the user's interest similarity is time-sensitive.

Overall, this paper presents a time-sensitive similarity calculation model that introduces user interest stability.

As following:

 （8）

Among them, ，， and  stand for the decentralized variance, for user  and user  reviews,  is the median of variance, respectively. Draw a conclusion in Figure 7 (x-axis is User ID, y-axis is User Score Variance), most of variance are located in [0.5, 1.5]. Therefore, , is the parameter derived from the experimental, and.  is the intersection items set for user  and user .  and  are the review time for user  and user  to item . stand for the parameter derived from experimental, and .  and  represents the sum of squares of user and user score on the item.。The model in the above formula (8) called the Stability\_Time\_Sim model

，STS in brief.

**2.3 The fusion of two Similarity Models**

In this paper, we introduce the advantages of the two similarity models, and the similarity model of user interest based on item popularity can effectively alleviate the problem of item bias. By introducing the similarity model of time-sensitive user interest stability, it can catch user interest in real, digging long tail items; improve the recommended system of novelty. So in order to make the recommendation better, we will linearize the two models, put forward the function  model:

（9）

This is  Model, which comprehensively put the stability of users, time-sensitive, and items popularity factor and forth into consider. Synthesize these factors to model. And the experiment shows that, the weighted similarity model has a marked improvement in the recommended quality.

**3. Experiment Design and Result Analysis**

**3.1 Experimental data set**

The dataset used in this lab is the data set provided by the MovieLens site developed by the GroupLens team at the University of Minnesota. MovieLens was founded in 1997 and is a Web-based recommendation system. Currently, the site offers three different levels of data sets: 100,000 records of 943 users to 1682 movies; 6040 users to 3900 films scores of 1 million data; 10 million scoring data made by 71567 users of 10,681 films.

This Experiment, the first case data sets above is used, in which each user evaluated 20 films at least. The sparse grade for this data sets is that, . In this paper, the experimental data set is divided into training set and test set, of which 80% of the training set, test set accounted for 20%.

**3.2 Evaluation Standard**

The Evaluation Standard for Quality of Recommended used in this paper is, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) respectively [19].

MAE as following:

 (10）

RMSE as following:

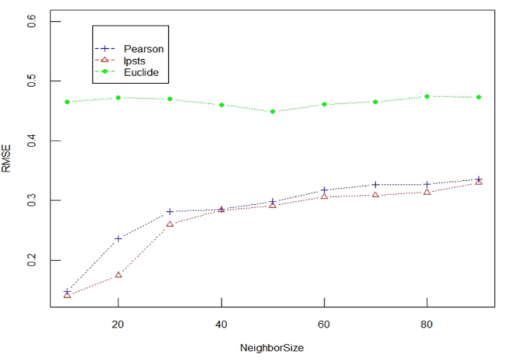
 (11)

**3.3 Experimental Results**

In this paper,

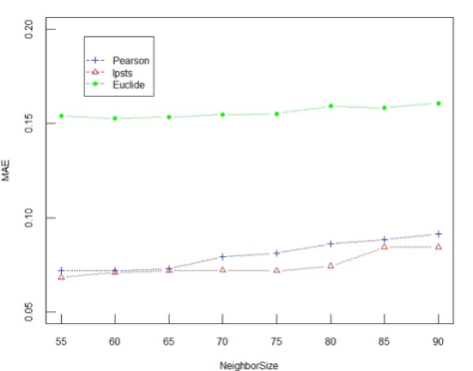
it make a comparison among the model, the Pearson model and the Euclide model. In the first case, w hen the parameters and parameters are fixe, as the number of neighbors increases, Check the effect of the three models on RMSE and MAE evaluation criteria. Another case is to fix the number of neighbors， and examine the effect of the three models on the evaluation criteria of RMSE and MAE, when the parameter and parameter and the parameter are gradually increasing according to the corresponding step size.

**3.3.1**  **Model Result**



**Figure 1** IPSTS Model, Pearson Model and Euclide Model in RMSE Comparison

Figure 1 (abscissa is the number of selected neighbors，ordinate is RMSE) shows the effect of each similarity model on RMSE when selecting different neighbors. The number of neighbors is 10, 20, 30, 40, 50, 60, 70, 80 and 90 severally. From such an analysis， know that the RMSE derived from  Model is less than the RMSE derived from Euclide Model and Pearson Model.(e.g. When the number of neighbors is 20, The RMSE of  Model about 6% lower than Pearson Model, and approximately 30% lower than the Euclide Model). Therefore, the error is reduced and the recommended quality is improved.

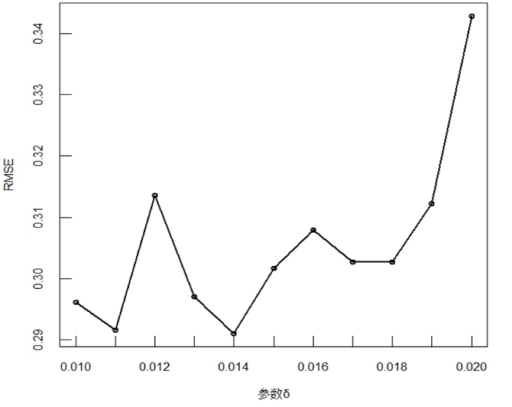


**Figure 2** IPSTS Model, Pearson Model and Euclide Model in MAE Comparison

Figure 2 (abscissa is the number of selected neighbors，ordinate is MAE) shows the effect of each similarity model on MAE when selecting different neighbors. The number of neighbors is 55，60，65，70，75，80，85 and 90 severally. From such an analysis， know that the MAE derived from IPSTS Model is less than the RMSE derived from Euclide Model and Pearson Model.(e.g. When the number of neighbors is 80, The RMSE of IPSTS Model about 1% lower than Pearson Model, and approximately 8% lower than the Euclide Model). Therefore, the recommended quality is improved.

### **3.3.2 The Influence of Parameter to** **Model**

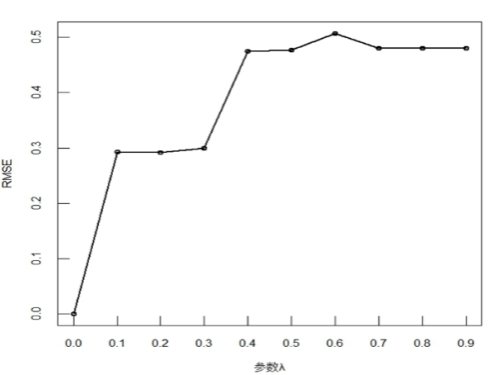
In the experiment, the selection of parameters is very important, in order to achieve satisfactory results of the experiment, while fully testing the robustness and applicability of the algorithm, more optimal algorithm, this section of the selection of the value of  the experiment, as far as possible to minimize the interference of other factors, only to test the impact of the parameterto model , we take the number of neighbors is 50, the parameterstake 0.2, the test with the increasing of, the model in the RMSE evaluation criteria on the effect.



**Figure 3** Parameterto IPSTS Model

### **3.3.3 The Influence of Parameter toModel**

At the same time, for examining the effect on weight parametertoModel combined with the result derived from above section 3.3.2. Let’s take the number of neighbors is 50, the parameterstake 0.014, the test with the increasing of, the model in the RMSE evaluation criteria on the effect.

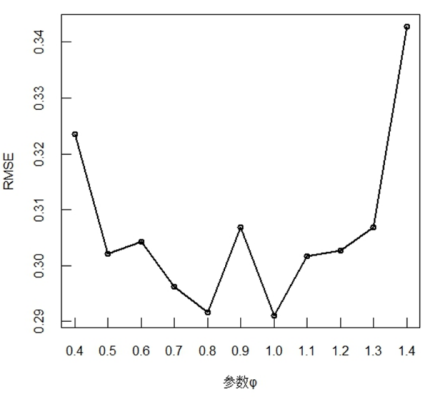


**图4** Parameterto IPSTS Model

Figure 4 (abscissa is  and ordinate is MAE) shows that, after= 0.1, with the increasing of, RMSE decreases initially and increases afterwards, and it will decrease again when it reach to a specific value. Later, with the increasing of, RMSE basically tend to smooth-out. In Figure 4，whenis 0.2, RMSE is the minimum, so in Model, parameter=0.2 is the optimization.

### **3.3.4 Parameter** **on** **Model**

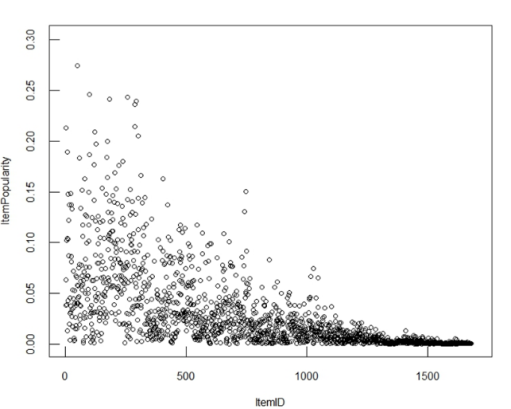
Checking the effect of parameter to  Model, with the result derived from above and the fix value, the number of neighbors is 50，=0.2，=0.014.



**Figure 5** Parameterto IPSTS Model

Figure 4 (abscissa is  and ordinate is MAE) shows that, when=1.0 the value of RMSE is Min. Therefore, in  Model, =1.0 is the optimization.

**3.3.5**  **Model on long tail items**



**Figure 6**  Diagram for Item ID and Item popularity

Figure 6 illustrates that these items exist long tail phenomenon. Among them, the more the item ID, the higher the prevalence, ItemID after 1000 are the long tail items. To test the model of the long tail items mining capacity, the experiment is as follows: Make a random selection of 3 users and make top-5 recommendation for them (recommend first 5 items). Users ID are 80,800,888. UserID is the user number that needs to be recommended, RecommenderSize = 5 is the recommended list size, NeighborSize = 90 (The number of neighbors is NeighborSize). In the Pearson model, RMSE = 0.33560, RMSE = 0.32965 in  model.

Table 1. When userID is 80, Item recommended of Pearson Model and IPSTS Model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item ID | | | | | |
| Pearson | 113 | 945 | 224 | 30 | 408 |
| IPSTS | 634 | 1467 | 1189 | 114 | 115 |

Table 1 ( ItemID is item serial number ) shows that, using  Model, decreases RMSE, at the same time improve the quality of recommendation, it also could dig out the long items which serial number are 1467, 1189 respectively. It eases the long tail phenomenon, and brings the surprise as well.

Table 2. When userID is 800， Item recommended of Pearson Model and IPSTS Model

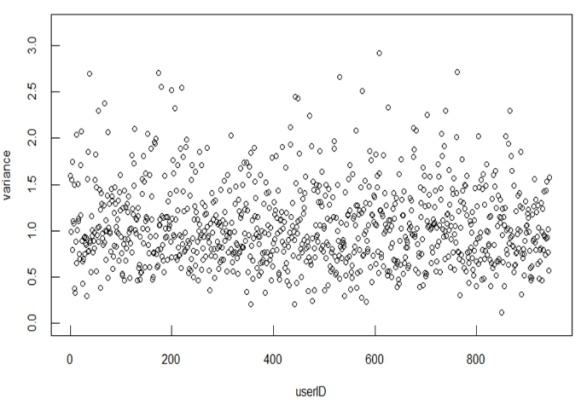
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item ID | | | | | |
| Pearson | 113 | 30 | 656 | 493 | 64 |
| IPSTS | 1159 | 1121 | 524 | 1103 | 114 |

Similarity, Table 2 ( ItemID is item serial number ) illustrate, it do not dig out the long tail items with using Pearson Model, while Model decreases RMSE and dig out three items whose ItemID are 1159，1121，1103 for making recommendation. It not only solves the long tail issue someway, but also gives users a sense of freshness.

Table 3. When userID is888 Item recommended of Pearson Model and IPSTS Model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item ID | | | | | |
| Pearson | 641 | 30 | 113 | 522 | 656 |
| IPSTS | 1467 | 1368 | 1512 | 615 | 169 |

Table 3 ( ItemID is item serial number ) figure out, when making recommendations to user whose ID is 888,  Model at the base of decreasing RMSE, give a recommendation strategy: 1467, 1368, 1512, 615, 169. There are 3 long tail items within those, However, Pearson Model neither give a lower RMSE compared with Model, nor dig out the long tail items on the recommend strategy list. Therefore, Model could ease the long tail phenomenon, and dig out the long tail for making a recommendation for users.



**Figure 7**

User ID, User assess scores variance scatter diagram

**4 Summary**

Recommend system is in the situation of that user do not have clear requirements to help user to find information that they are interested in from a large amount of data, then display and recommend to user by appropriate ways. This essay focus on reducing grading error and excavating long tail items, relieving pop item polarization problem and import interest stability, at the same time, which also focus on time sensitive factor thereby, to construct a time sensitive similarity fusion model of importing interest stability.

**Reference**

1. 佚名. 第36次《中国互联网络发展状况统计报告》[J]. 网络传播, 2015(7):54-59.
2. Resnick P, Iacovou N, Suchak M, et al. GroupLens: an open architecture for collaborative filtering of netnews[J]. Working Paper Series, 2015:175--186.
3. 邢春晓,高凤荣,战思南.适应用户兴趣变化的协同过滤推荐算法[J].计算机研究与发展,2007.44(2): 296-301.
4. 项亮. 动态推荐系统关键技术研究[D].北京:中国科学院自动化研究所,2011.
5. Liu B, Wu Y, Gong N Z, et al. Structural Analysis of User Choices for Mobile App Recommendation[J]. Acm Transactions on Knowledge Discovery from Data, 2016, 11(2):17.
6. Rashid A M, Lam S K, Karypis G, et al. ClustKNN: a highly scalable hybrid model-& memory-based CF algorithm[C]// The Workshop on in Proc of Webkdd. 2006.
7. Elkahky A M, Song Y, He X. A Multi-View Deep Learning Approach for Cross Domain User Modeling inRecommendationSystems[C]//TheInternationalConference. 2015:278-288.
8. Wang H, Wang N, Yeung D Y. Collaborative Deep Learning for Recommender Systems[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2014:1235-1244.
9. Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, Usa, August. 2008:426-434.
10. Kim D, Yum B J. Collaborative filtering based on iterative principal component analysis[J]. Expert Systems with Applications An International Journal, 2005, 28(4):823-830.
11. Ahn H J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem[J]. Information Sciences, 2008, 178(1):37-51.
12. Liu H, Hu Z, Mian A, et al. A new user similarity model to improve the accuracy of collaborative filtering [J]. Knowledge-Based Systems, 2014, 56(3):156-166.
13. Li S, Kawale J, Fu Y. Deep Collaborative Filtering via Marginalized Denoising Auto-encoder[C]// The, ACM International. 2015:811-820.
14. Wu Y, Dubois C, Zheng A X, et al. Collaborative Denoising Auto-Encoders for Top-N Recommender Systems[C]// ACM International Conference on Web Search and Data Mining. ACM, 2016:153-162.
15. Strub F, Mary J, Gaudel R. Hybrid Recommender System based on Autoencoders[J]. 2016:11-16.
16. Wang X, Lu W, Ester M, et al. Social Recommendation with Strong and Weak Ties[C]// The, ACM International. 2016:5-14.
17. Vuurens J B P, Larson M, De Vries A P. Exploring Deep Space: Learning Personalized Ranking in a Semantic Space[J]. 2016:23-28.
18. Adomavicius G, Kwon Y O. Improving Aggregate Recommendation Diversity Using Ranking-Based Techniques[J]. Knowledge & Data Engineering IEEE Transactions on, 2012, 24(5):896-911.
19. Herlocker J L. Evaluating collaborative filtering recommender systems[J]. Acm Transactions on Information Systems, 2004, 22(1):5-53.