

RESEARCH ON PERSONALIZED RECOMMENDATION ALGORITHM FUSING TIME AND LOCATION

Zhongyong Fan¹, Yongqian Zhao², Yongkang Wang², Zhijun Zhang^{2*}

¹Rizhao Polytechnic, Shandong, Rizhao, China

²School of Computer Science and Technology, Shandong Jianzhu University, Fengming Road, Jinan, China

Corresponding Author.

Abstract

With development of recommendation systems, they are faced with more and more challenges. In order to relieve problems existing in commodity selection by users of different preferences from different regions, personalized recommendation based on location information has emerged. Nowadays most recommendation systems based on location information neglect the fact that users' preference will change with time. To solve the above problem, geographic location and time factor of users are effectively combined in this paper, and a personalized recommendation algorithm TLP combining time and location information is proposed. This algorithm determines the users' geographic location according to postcode information of the users, uses pyramid quadtree model to distribute users into nodes at each layer in the pyramid, utilizes collaborative filtering algorithm for local recommendation in each node, introduces a time function to regulate time-dependent change of user interests when calculating user similarity at each node and finally realizes a comprehensive recommendation by distributing a weight for recommendation result at each layer in the pyramid quadtree. A comparative experience is carried out for recommendation performance of this algorithm on MovieLens dataset, and experimental results indicate that this algorithm is of better recommendation effect.

Keywords: Personalized Recommendation, collaborative Filtering, location Information

I. Introduction

With rapid development of ever-changing computer networks and e-commerce systems, commodities available to be selected by users are abruptly increasing. It's more and more difficult for users to acquire commodities of their interests from the sea of commodities. How to present commodities in front of users who are interested in them has become a current research hotspot, and personalized recommendation is an important tool emerging to solve this contradiction. Personalized recommendation makes it possible for users to find commodities of their interests according to their own historical behaviors and present commodities in front of users who are interested in them so as to realize a win-win pattern between commodity consumers and commodity providers.

As an important consumer good in people's daily life, the movie has been a hot application field of personalized recommendation and it can help users to highly efficiently find works of their interests in the vast movie video database. Netflix and Douban are representative products of application of personalized recommendation. Many movie recommendation systems adopt collaborative filtering algorithm namely finding nearby users according to users' watching historical records and then recommending according to predicted ratings. Few of them have considered geographical location information of users, users from different regions have different preferences in

reality, and for instance, French ethnic group in Canadian Quebec Province has totally different user preference from Toronto. Movie preference of Spanish ethnic group in southwestern America is approximate to some user groups in northern Mexico. Meanwhile, users' preferences for commodities will drift with time. When new commodities emerge, commodity popularity will also experience continuous change. Based on the above considerations, personalized recommendation algorithm TLPR combining time and location information was proposed in this paper, which could obviously improve recommendation effect.

Section 2 introduces related work conducted for recommendation relating to time information and location information in this paper. Section 3 mainly introduces the personalized recommendation algorithm combining time and location information. Section 3 conducts a comparative experimental analysis of different algorithms and verifies advancement of this algorithm. Section 5 summarizes and expected future research work based on this algorithm.

II. Related Work

Collaborative filtering algorithm, which was proposed by Typest, is a recommendation algorithm [1-5] which has been most extensively applied at present. This algorithm obtain some recommendation results to the users according to users' purchase record, scoring record, browser page rolling time and other information. Many scholars have made different improvements based on the collaborative filtering algorithm in order to recommendation precision of the recommendation algorithm and favorable recommendation effects have been achieved. Recent studies find that user behavioral pattern can affect recommendation effect to a certain degree. For example, new users and old users have different selection patterns, user preference will change with the age, and so will their interests be in different periods and in different places. Therefore, digging user behavioral pattern can realize recommendation under complicated conditions and improve recommendation precision. On this basis, domestic and overseas scholars have found that space-time statistical characteristics of user behaviors can also be used to improve recommendation or design applications for specific scenes.

2.1 Study of personalized recommendation algorithm combining time factors

In fact, it's simply assumed that user interest will present exponential decrease with time, and improved recommendation effect can also be obtained [6, 7]. The study[8] carried out by Sugiyama et al. put forward a time-based collaborative filtering technology through detailed analysis of browsing history of the user within one day. In Literature [9], the author put forward a time-based attenuation function to process time series data according to clustering of each user and each term, and this function could calculate corresponding time weights for different resources. Literature [10] proposed calculating user ratings according to term release time, user purchasing time and information combining the two. This temporary information could be used to improve recommendation precision in a dynamic e-commerce environment. In order to solve sparsity problem, Wu et al. [11] put forward a collaborative filtering algorithm based on user features and time effect, and introduced time information and user information when calculating the similarity so as to improve similarity calculation and reduce sparsity problem. Based on the collaborative filtering algorithm, Wu et al. [12] combined time factor and user ratings to dynamically distribute weight of each rating, and relieved sparsity problem by using predicted ratings to fill similarity matrix and a certain effect was achieved.

2.2 Study of personalized recommendation algorithm combining location factors

Not only time will influence change of user interest but also location change will also generate a great influence. With rapid development of the internet and development and popularization of GPS and other mobile phone locating technologies, location-based service has become a problem attracting extensive concern in the academic

circles and within the industry. Location information-based recommendation has become a research hotspot and important application scene. As for the influence of location information on the recommendation system, Google pushed out a service called Hotpot in 2010. This service lets the user rating the place where he has been and then recommends places to the user according to the rating. Spanish telecommunications personnel once designed a location-based movie recommendation system and provided a detailed technical report [13]. Scientific research workers from University of Minnesota put forward a recommendation model called LARS and closely related to the user place [14]. Yuan et al. [15] proposed layered exploration to measure similarity based on user geographic location on location-based service platform, and this method could effectively identify users with similar geographic visit locations; Hu et al [16] combined user behavior trace and urban functions in social media so as to identify user role information and provide a method of calculating user similarity in the recommendation system. Liang et al. [17] designed an index tree applied to personalized recommendation and based on location coding and applied the index structure to personalized recommendation in an innovative way. Li et al. [18] founded group patterns with similar behavior traces under unequal time interval constraints and solved the problem of the traditional trace group pattern digging algorithm which could only process GPS data with fixed time interval sampling constraints.

2.3 Study of personalized recommendation algorithm combing time and location

Yuan et al. [19] put a POI (Point of Interest) recommendation system sensitive to time, which combined geographic location and time information to recommend places where the users had not been. Li et al. [20] put forward a brand-new personalized location recommendation model for LBSN, and this model included influence of social contact and space-time characteristics, which reduced limitations caused by matrix sparsity and cold boot problem to recommendation performance to a certain degree and improved location recommendation precision, but the emphasis was laid on exploration into social relationship while neglecting the relationship between locations. Ma et al. [21] put forward a multi-factor-combined personalized location recommendation algorithm by combining geographic location information and user relation and according to the obtained main causes influencing user moving behaviors. This algorithm has effectively combined user preference information, influence of social relations, present user location, time interval and other multiple factors, which can improve location recommendation precision.

III. Recommendation Algorithm TLPR

3.1 Traditional user-based collaborative filtering recommendation model

The most basic and simplest algorithm in the recommendation systems is user-based collaborative filtering recommendation algorithm, based on which many recommendation algorithms are derived. Its implementation mainly includes two steps.

(1) Seek for the users with similar preferences to the user to which the recommendation will be made by calculating similarities between users.

There are many computational methods of similarities, here cosine similarity calculation is adopted, and the formula is as follow:

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

$N(u)$ is item set for which user u has behaviors and $N(v)$ is item set for which user v has behaviors.

(2) Find item favored by similar users and not contacted by the user to which the recommendation will be made and recommend them to the user.

After interest similarities between users are obtained, UserCF algorithm will recommend items liked by K users with similar interests to the user for recommendation. The following formula (2) used to measure interest degree of user u in item i :

$$p(u, i) = \sum_{v \in S(u, K) \cap N(i)} w_{uv} r_{vi} \tag{2}$$

Where $S(u, K)$ includes K users with approximate interests to user u ; $N(i)$ is user set with behaviors for item i ; w_{uv} is interest similarity between user u and user v ; r_{vi} is interest of user v in item i .

3.2 Personalized recommendation algorithm combining time

Traditional recommendation algorithms have neglected influence of time on user interest change, which affects predicting precision to a certain degree. User-based and time context-dependent collaborative filtering algorithm is as follow.

A time attenuation function is introduced when calculating user similarity.

$$f(|t_{ui} - t_{vi}|) = \frac{1}{1 + \alpha |t_{ui} - t_{vi}|} \tag{3}$$

Where t_{ui} is the time for user u to generate behaviors for item i ; t_{vi} is the time for user v to generate behaviors for item i .

User-based collaborative filtering algorithm was introduced in 3.1. After time information is obtained, the following improvements can be made for similarity calculation.

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

The greater the time difference between user u and user v to generate behaviors for item i , the smaller the interest similarity between the two users. Besides consideration of the influence of time on similarities, influence of time information on predictor formula should also be taken into consideration. Generally speaking, present user behaviors should have greater relations with recent behaviors. Therefore, the predictor formula can be corrected in the following way:

$$p(u, i) = \sum_{v \in S(u, K)} w_{uv} r_{vi} \frac{1}{1 + \alpha (t_0 - t_{vi})} \tag{5}$$

Where t_0 present time and α is time attenuation parameter. The above formula indicates that: the closer the t_{vi} is to t_0 , the higher ranking the item similar to item i will obtain in recommendation list for user u .

3.3 Time and location-based personalized recommendation algorithm

The user can be distributed to a node of one series using pyramid quadtree model according to user postcode information and geographic information, and each node includes behavioral dataset of users at the same location

with the user at this layer. Then local recommendation is implemented in each node using the time-based collaborative filtering algorithm, and finally a weight value is distributed to recommendation result at each layer in the pyramid quadtree for comprehensive recommendation. Besides recommending to ordinary users, this model processes particularity of boundary users in two regions so that the recommendation can be more accurate.

3.3.1 Pyramid quadtree model

Pyramid quadtree structural model was adopted in this paper to conduct regional division of the dataset as shown in Fig. 1. The model divides the region to be studied into H layers (tree height). For tree height h, the whole space is divided into $4^h - 1$ node grids with the same size, and a user-based collaborative filtering model is built according to historical evaluation information between users for an item. A user and his evaluation of related items will appear in each layer among H layers and both of this user and his evaluation information have participated in establishment of the collaborative filtering model among the top-down node grids. Generally speaking, the pyramid quadtree model will be relatively stable, and only when a large batch of new data are introduced will the influence be generated on results of the collaborative filtering model for tree nodes.

During the concrete recommendation, regional division of the users is implemented from layer 0 downward, collaborative filtering of each layer will generate a local recommendation results list, and the final output result is the comprehensive measurement of all of the previous local results.

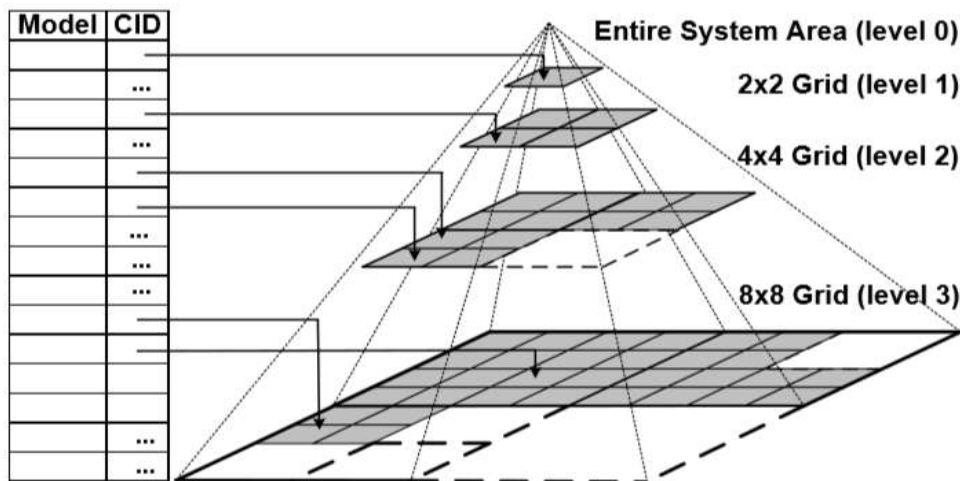


Fig. 1 Schematic diagram of pyramid model

3.3.2 Time and location-based recommendation model

Personalized movie recommendation for users is largely implemented in three steps.

Step 1: this step starts from root nodes of pyramid quadtree. For each user u_i to be divided in the user dataset (User), the regions where the users are located are subdivided according to the first 1, 2 and 3 figures of their postcodes until leaf nodes.

Step2 is the collaborative filtering step. For each node i in the set, traversal of each user u_j and his ratings for some movies is implemented, and these information are used to construct a user-movie grading sheet as Table 1.

Table 1 User-movie grading sheet

	Movie ₁	Movie2	Movien
u1	3	4	5

u2	2	4	3
.....			4
um	2	1	5

Next is calculation of user similarities, during which influences of time factors on user degree of interest should be considered, and time factors should be included in similarity calculation. For the user u_i for recommendation, K users with the highest similarity to user u_i are found. In each candidate node, the list of P movies ($movie_1, movie_2, \dots, movie_p$) recommended by the K users to user u_i according to predicting ratings in an ascending order is recorded.

Step 3 is the link of the final result generated by weighted recommendation. Several local recommendation lists (List 1, List 2, ... , List q) are generated by several nodes in Step 2. Weight coefficients will be distributed from top layer gradually until leaf nodes. When the height of the pyramid quadtree model is h, ratings of one Mov_i in List 1 ,List 2,..., List q are set as $R_1, R_2, R_3, \dots, R_n$ respectively (when this movie appears in one List, the rating is 0). The following formula is used to calculate final weighted recommendation rating $R (Mov_i)$ of one movie.

$$R(Mov_i) = R_1 \times \beta_1 + R_2 \times \beta_2 + \dots + R_n \times \beta_n \tag{6}$$

Where β_i is a coefficient between 0 and 1, and $\beta_1 + \beta_2 + \dots + \beta_n = 1$ and $\beta_1 \leq \beta_2 \leq \dots \leq \beta_n$ are satisfied. The coefficient is obtained through an experiment in the recommendation process. Through multiple groups of experimental comparison, it's found that when $h=4$, the coefficient is taken as 0.2, 0.35 and 0.45 respectively; when $h=5$, the coefficient is taken as 0.15, 0.2, 0.25 and 0.4 respectively, and favorable recommendation effects can be achieved under both circumstances. Finally the recommendation list Top N is obtained.

The model algorithm is as follow:

Input: User set, the user u_i for recommendation, and depth h of the pyramid model.

Output: final weighted grading matrix.

- 1 . Traversal of nodes experienced by the user is implemented and recorded in Path set
- 2 . for($u_i \in$ User with postcode)
- 3 . Find_Path(u_i)->Path[u_i][]
- 4 . For(Node_i \in Path[u_i][])
- 5 . For($u_2 \in$ Node_i)
- 6 . Time factors are added in the calculation of user similarities
- 7 . Similarity_distance(prefer, $u_i, u_j, f(|t_{ui} - t_{vi}|)$)
- 8 . Collaborative filtering is implemented for users layer by layer
- 9 . CF_Model (u_i) ->Result:List I [$movie_1, movie_2, \dots, movie_p$]
10. List=[List1, List2, ... List n],
11. Choose the weight $\beta_1, \beta_2 \dots \beta_n$
12. Weighted recommendation, K is returned number of movies
13. Recommendation-List(u_i) {
 - For(List_i \in List)
 - If($movie_i \in$ List_i)
 - $R(movie_i) = R(movie_i) + R(List_i) \times \beta_i$
 - Add R ($movie_i$) to Ratings []
 - Return Ratings [0: K]

}

IV. Experiment and Analysis

4.1 Introduction of the dataset

The dataset used in this paper is MovieLens. This dataset contains three different versions; this paper mainly carries out comparative experiments in two versions of 100K and 1M data sets. The dataset mainly contains three parts: ratings. Data, user.data and movie.data. Ratings Data includes user id, movie id and ratings made by users on corresponding films and timestamps. In addition, user and movie-related information are saved in user.data and movie.data respectively, the former records user id, gender, age, occupation, postcode, etc., and the latter records movie id, movie name, movie type, etc.

In order to ensure precision of experimental results and exclude the influence caused by accidental factors. Two data sets of different sizes were evenly and randomly divided into 5 portions, the one portion is taken as later-stage test set and the residual 4 portions are used as training sets. Meanwhile, in order to ensure overfitting phenomenon of assessment indexes, N experiments should be carried out, different test sets are used in different experiments, and finally mean value of these experimental results is taken as the final index.

4.2 Evaluation indexes

Recommendation result in this experiment is Top N recommendation, corresponding evaluation criterion is rating criterion of the recommendation results, and predicting precision of Top N recommendation is measured through precision and recall.

Definition of the recall (also called recall ratio) of the recommendation system is as follow:

$$\text{recall} = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|} \quad (7)$$

Definition of the precision of the recommendation system is as follow:

Where $R(u)$ is the list generated for the users through inductive learning of the obtained model according to user behaviors on the training set; $T(u)$ is behavioral list of users on the test set.

F1-rating is defined as harmonic mean of precision and recall, which can comprehensively reflect various properties of a recommendation algorithm. It has been extensively applied in the studies of recommendation algorithms. Generally speaking, the higher the F1-rating value, the better the algorithm performance.

4.3 Experimental results and analysis

The recommendation model combining time and location factors (abbreviated as TLPR) proposed in this paper is evaluated through an experiment, and it's compared with user-based collaborative filtering (hereinafter abbreviated as User-CF), recommendation algorithm combining time factors (hereinafter abbreviated as T-CF) and recommendation algorithm combining location factors (hereinafter abbreviated as ADPR) in aspect of their recommendation effects.

In order to test algorithm performance, for a given user u , a recommendation list with length being K is generated

for the user using the data in the training set. Algorithm performances are compared when different K values are taken.

First, we need to determine the number of layers of the Pyramid model with the best recommendation. When the length of the recommended list is K, The number of layers in Pyramid is determined by the recommended effect of different algorithms in different layers of Pyramid model. The four algorithms involved in the comparison, TLPR, ADPR, User-CF, and T-CF, only use the Pyramid model in the two kinds of TLPR and ADPR, and the Precision of the two algorithms in the Pyramid model of different layers is shown as shown in the Fig. 2 and Fig. 3.

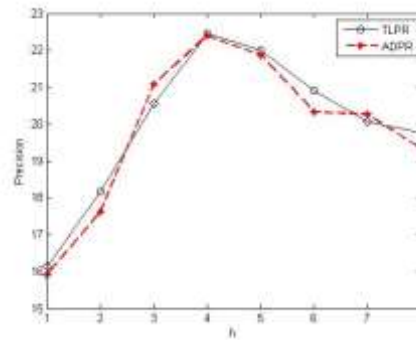


Fig 2 the impact of H in the 1M data set

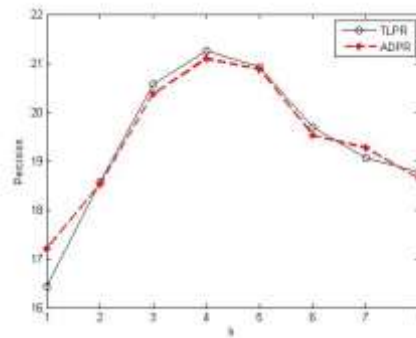


Fig 3 the impact of H in the 1M data set

It is intuitive to see that when h=4 and h=5 are used, the algorithm is the best.

In the 1M version of the movieLens data set, 1,000 users are selected as candidate users for recommendation. The final result is mean value of evaluation indexes of the 1,000 users, and results are shown in Table 2.

Table 2 Comparison of evaluation indexes of four recommendation algorithms under 1M dataset

K	TLPR		ADPR		User-CF		T-CF	
	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)
10	14.323	20.873	14.457	20.541	12.725	19.602	13.433	19.846
15	15.402	21.766	15.378	21.366	15.308	20.415	15.407	20.973
20	15.585	22.041	15.496	21.795	16.109	20.369	16.305	21.064
25	18.237	22.967	17.872	22.551	15.051	21.325	15.756	21.475
30	18.216	22.860	18.033	22.399	17.231	22.055	17.856	22.078
35	19.764	23.971	19.608	23.305	17.112	21.766	18.021	22.158

In the 100K version of the MovieLens data set, 300 users are selected as candidate users for recommendation. The final result is mean value of evaluation indexes of the 1,000 users, and results are shown in Table 3.

Table 3 Comparison of evaluation indexes of four recommendation algorithms under 100K dataset.

K	TLPR		ADPR		User-CF		T-CF	
	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)
10	13.226	19.136	13.426	19.247	12.772	19.124	13.612	19.336
15	14.402	20.252	14.308	19.856	14.425	19.726	14.375	19.602
20	15.185	20.867	15.076	20.576	15.442	20.226	15.842	20.476
25	16.856	22.015	16.645	21.766	15.785	20.960	15.874	21.021
30	17.220	22.023	17.148	21.878	16.725	21.072	16.886	21.477
35	18.134	22.625	17.976	22.356	16.974	21.756	17.126	21.896

The table lists recall and precision data of four different recommendation algorithms under 1M dataset and 100K dataset. For the sake of intuitive data analysis of this table, broken line graphs of recall and precision and histogram of F1-rating are respectively given.

Intuitive broken line graphs of recalls of the four different algorithms are shown in Fig. 4 and Fig. 5, respectively being TLPR, ADPR, User-CF and T-CF from up to bottom. As shown in the graphs, it can be seen that as K value increases, overall TLPR value of the recall is the highest, and when K=15, index differences among the four algorithms are not great. T-CF performance is the best when K=20.

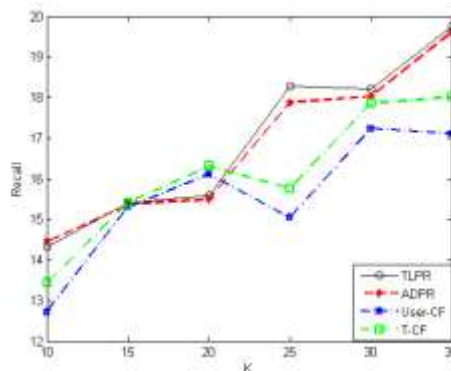


Fig. 4 Recall comparison under 1M data set

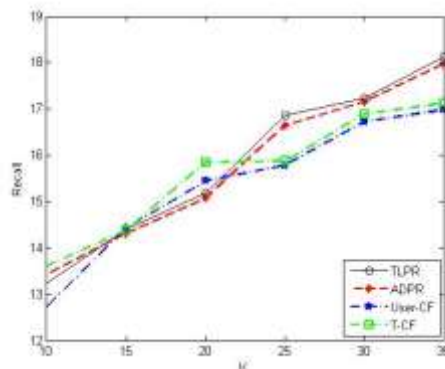


Fig. 5 Recall comparison under 1M data set

Intuitive broken line graphs of the precision values of the four algorithms are shown in Fig. 6 and Fig.7, respectively being TLPR, ADPR, User-CF and T-CF. It can be seen that as K value increases, TLPR index is obviously superior to the other three algorithms.

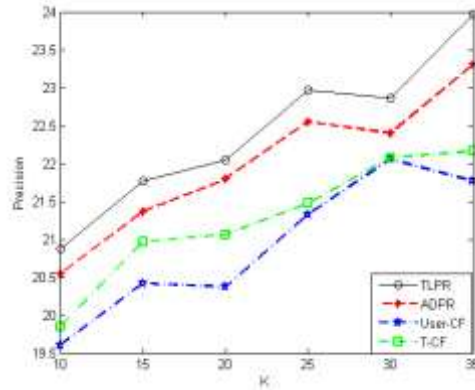


Fig. 6 Precision comparison under 1M data set

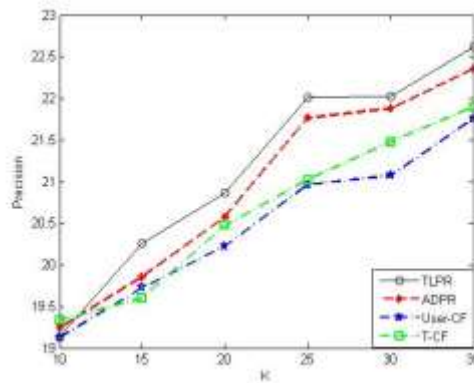


Fig. 7 Precision comparison under 1M data set

F1-rating indexes of the four algorithms are shown in Fig. 8 and Fig.9. It can be seen that as K value increases, overall F1-ratings of four algorithms present rising trend, and F1-rating of TLPR algorithm is obviously higher than other three algorithms.

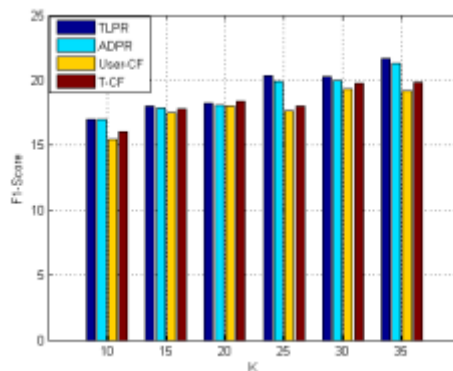


Fig. 8 F1-rating comparison under 1M data set

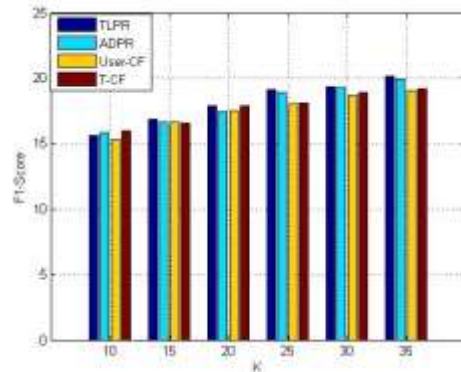


Fig. 9 F1-rating comparison under 1M data set

It can be seen from the intuitionistic graph of each index: parameter K is an important influence factor of the algorithms, and its change will affect various indexes of the recommendation algorithms to different degrees. The larger the amount of data, the better the indicators of each algorithm. On the whole, as K value increases, overall indexes of the algorithms all present rising trend, but they don't present strict linear relations. For example, when K value is 30, precision of User-CF algorithm is the highest, so K selection and size of data are of great importance to improvement of the algorithm performance. Experimental results indicate that under the same circumstance, the recommendation effect is the better when time and location factors are taken into consideration simultaneously.

V. Conclusions

Compared with the traditional recommendation algorithms, the recommendation algorithm combining time factors and geographic location factors includes more information which can be used to establish a recommendation model with better recommendation effect. The emphasis in this paper was laid on influences of time and location on user preferences, and a recommendation model combining time and geographic location factors was designed and implemented. Through an experimental comparison, the model has better recommendation effect.

It's recognized that time and geographic location have great influences on performance of the recommendation algorithm in this paper and recommendation performance is improved through the experiment, but there are still some deficiencies with improvement space. For example, when the data size is large, the time spent to search and collaborative filtering will be very long; taking postcodes as territorial division signs has neglected change of user interest with the change user location.

Acknowledgments

This paper is made possible thanks to the generous support from the Key Research and Development Program of Shandong Province (2018GGX106006), Jinan Science and Technology Project (201704065), A Project of Shandong Province Higher Educational Science and Technology Program (J17KA070), PhD Funding Project of Shandong Jianzhu University (No. X19044Z), Science and Technology Program Project of Shandong Colleges and Universities (No.J17KA070), Undergraduate Education Reform Project of Shandong Province (No.Z2016M016, Z2016M014, M2018X197).

References

- [1] Aligon J, Gallinucci E, Golfarelli M, et al. "A collaborative filtering approach for recommending

- OLAP sessions”. *Decision Support Systems*, 69(C), pp. 20-30, 2015.
- [2] Liu H, Hu Z, Mian A, et al. “A new user similarity model to improve the accuracy of collaborative filtering”. *Knowledge-Based Systems*, 56(3), pp. 156-166, 2014.
- [3] Sánchez-Moreno D, Gil González A B, Muñoz Vicente M D, et al. “A collaborative filtering method for music recommendation using playing coefficients for artists and users”. *Expert Systems with Applications An International Journal*, 66(C), pp. 234-244, 2016.
- [4] Deng A L, Zhu Y Y, Shi B L. “A Collaborative Filtering Recommendation Algorithm Based on Item Rating Prediction”. *Journal of Software*, 14(9), pp. 54-65, 2003.
- [5] XiJun Liu, Zhou Tao. “Social dynamics”. Science Press, 2012.
- [6] J. Liu, G. Deng. “Link prediction in a user-object network based on time-weighted resource allocation”. *Physica A*, 39, pp. 3643-3650, 2009.
- [7] Y. Koren, Collaborative filtering with temporal dynamics, in: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM Press, New York, pp. 447-456, 2009.
- [8] Sugiyama K, Hatano K, Yoshikawa M. Adaptive web search based on user profile constructed without any effort from users[C]. *International Conference on World Wide Web. The Institute of Electronics, Information and Communication Engineers*, pp. 675-684, 2004.
- [9] Zhao Y, Zhang C, Zhang S. “A Recent-Biased Dimension Reduction Technique for Time Series Data”. 3518, pp. 751-757, 2005.
- [10] Tong Q L, Park Y, Park Y T. “An empirical study on effectiveness of temporal information as implicit ratings”. *Expert Systems with Applications*, 36(2), pp. 1315-1321, 2009.
- [11] WU Xiong-feng, JIA Nian. “A Collaborative Filtering Algorithm Based on User Characteristics and the Effects of Time”. *Modern computer*. (10), pp. 21-24, 2016.
- [12] WU Fei, YU La-sheng, TENG Mei. “A Collaborative Filtering Algorithm Based on time effect”. *Computer engineering and Science*, 39(11), pp. 2095-2101, 2017.
- [13] Geolocated recommendations. [Http: Xavier.amatriain.net/pubs/Geolocated Recommendations](http://Xavier.amatriain.net/pubs/Geolocated Recommendations). PDF.
- [14] Levandoski J J, Sarwat M, Eldawy A, et al. “LARS: A Location-Aware Recommender System”. *IEEE, International Conference on Data Engineering*. IEEE, pp. 450-461, 2012.
- [15] YUAN Shu-han, CHEN Wei-bin, FU Shun-kai. “User behavior similarity analysis of location based social network”. *Computer application*, 32(2), pp. 322-325, 2012.
- [16] HU Yahui, LI Shijun, YU Wei, YANG Sha, FANG Qiqing. “Inferring Social Roles with Spatio-temporal Awareness”. *Journal of electronics and information*, 38(3), pp. 517-522, 2016.
- [17] LIANG Junjie, GAN Wenting, YU Dunhui. “Personalized recommendation algorithm based on location bitcode tree”. *Computer application*, 36(2), pp. 419-423, 2016.
- [18] Xiao-Xu L I, Ya-Xin Y U, Zhang W C, et al. “Mining Coteries Trajectory Patterns for Recommending Personalized Travel Routes”. *Journal of software*. 29(3), 2018.
- [19] Yuan Q, Cong G, Ma Z, et al. “Time-aware point-of-interest recommendation”. *International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pp. 363-372, 2013.
- [20] Li Heng. “Social Context and Temporal-Spatial Data based Personalized Location Recommendation”. ChongQing University, 2016.
- [21] Ma Hui Yi. “The Study of Personalized Location Recommendation Algorithm Based on the Integration of Multi-Factor”. LanZhou University, 2016.