A Multi-attribute Collaborative Filtering Recommendation Algorithm Based on Improved Group Decision-Making

Changrui Yu, Yan Luo, Kecheng Liu

A Multi-attribute Collaborative Filtering Recommendation Algorithm Based on Improved Group Decision-making

Yu Changrui[1], Luo Yan[2], Liu Kecheng[1,3]

[1] School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai 200433, China
yucr@sjtu.edu.cn
2 The Sydney Institute of Language and Commerce,Shanghai University,20 Chengzhong RD, Shanghai 201800, China
luoyan@shu.edu.cn
3Informatics Research Centre, Henley BusinessSchool, University of Reading, Reading, RG6 3XA, U.K.
k.liu@reading.ac.uk

Abstract. The paper builds an evaluation model of user interest based on resource multi-attributes, proposes a modified Pearson-Compatibility multi-attribute group decision-making algorithm, and introduces the algorithm to solve the recommendation problem of *k*-neighbor similar users. Considering the characteristics of collaborative filtering recommendation, the paper addresses the issues on the preference differences of similar users, incomplete values, and advanced converge of the algorithm. Thus the paper realizes multi-attribute collaborative filtering. Finally, the effectiveness of the algorithm is proved by an experiment of collaborative recommendation among multi-users based on virtual environment.

Keywords: Personalized recommendation, Pearson-Compatibility, Group decision-making, Multi-attribute, Collaborative filtering

1 Introduction

The goal of a recommender system is to generate meaningful recommendations to users for items or products that might interest them [1]. In many markets, consumers are faced with a wealth of products and information from which they can choose. To alleviate this problem, many web sites attempt to help users by incorporating a recommender system that provides users with a list of items and/or web pages that are likely to interest them. There are real-world operations of industry strength recommender systems, for example the recommendations for books on Amazon, or movies on Netflix, and so forth.

As one of the most successful approaches to building recommender systems, collaborative filtering (CF) uses the known preferences of a group of users to make recommendations or predictions of the unknown preferences for other users[2]. The developers of one of the first recommender systems, Tapestry [1] (other earlier recommendation systems include rule-based recommenders and user-customization), coined the phrase “collaborative filtering (CF),” which has been widely adopted regardless of the facts that recommenders may not explicitly collaborate with recipients and recommendations may suggest particularly interesting items, in addition to indicating those that should be filtered out. The fundamental assumption of CF is that if users X and Y rate n items similarly, or have similar behaviors (e.g., buying, watching, listening), and hence will rate or act on other items similarly.

Literature [3] studies have shown that users' interest to a product or service are affected by user topic preferences, content preferences, user habits, public evaluation and other factors, and these factors is decided by the different attributes of items. For example, the reasons of users liking a new movie may be caused by one or more attributes of the movie, such as the director, star, theme, content, style, public comment and other factors. Thus, in the application of collaborative filtering algorithm, it is necessary to use multi-attribute analysis model, that the user rating to an item should be from a different perspective (attributes) to describe their interests preferences.

However, the current research work of multi-attribute collaborative filtering focus on clustering users and resources based on the attribute information, and the recommended method is still more traditional. Such methods can only obtain a set of potential interest items of target users, but the reasons of such a recommendation is not given to the target user. In addition, the present study do not consider the characteristics differences of similar users interested in the item attributes. It can lead to recommendations deviation. For example, in the traditional way, User B is the most similar to the target user A, because A and B on the same film have the same degree of interest. But if they prefer the film properties are completely different, it will lead to recommendations deviation when we give greater weight given to B used to predict interest preferences of A.

We think that the multi-attribute collaborative filtering can be regarded as a group decision-making process. By building the rating matrix of target items for the similar users, we can remove the user who has a large attribute preference difference with target user from the nearest user set, and save the problem of recommendations deviation. And we can analyze the user's interest performance from the view of item's attributes, give the reasons descriptions for the recommendation. In order to achieve this goal, the paper has proposes a modified Pearson-Compatibility multi-attribute group decision-making algorithm, and introduces the algorithm to solve the recommendation problem of k-neighbor similar users. The organization of the paper is as follows. We review recommender systems and multi-attribute utility theory in Section 2.In section 3, with applied ontology method to describe user profile, we introduce detailed how to build a user interest model. In section 4, we expound the algorithm in each steps specifically. In section 5, an experiment is reported and the findings also be discussed.

2 Descriptions of Basic Model

A user's comment on a certain item is usually an integration of multi-attribute comments made from different angles. Suppose an item is shown as follows:

P = { $a_1, a_2, a_3, \ldots a_n$ }

Based on the revised rating model, the paper establishes the user rating matrix. Suppose the user set is denoted as U = { $U_1, U_2, \ldots U_p$ } and the user U_j rating for item P_i is denoted as $A(U_j, P_i)$:

$$
A(U_j, P_i) = \begin{array}{c c}
\begin{array}{cccccc} a_1 & a_2 & a_3 & & a_{n-1} & a_n \end{array} & \\
\begin{bmatrix}
\omega_{11} & \omega_{12} & \omega_{13} & \cdots & \omega_{1n-1} & \omega_{1n} \\
\omega_{21} & \omega_{22} & \omega_{23} & \cdots & \omega_{2n-1} & \omega_{2n} \\
\omega_{31} & \omega_{32} & \omega_{33} & \cdots & \omega_{3n-1} & \omega_{3n} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\omega_{(n-1)1} & \omega_{(n-1)2} & \omega_{(n-1)3} & \cdots & \omega_{(n-1)(n-1)} & \omega_{(n-1)n} \\
\omega_{n1} & \omega_{n2} & \omega_{n3} & \cdots & \omega_{n(n-1)} & \omega_{nn}
\end{bmatrix} &
\begin{array}{c} a_1 \\ a_2 \\ a_3 \\ \cdots \\ a_{n-1} \\ a_n \end{array}
\end{array}
$$

Where ω_{xy} is the importance of attribute a_x of product P_i in comparison with attribute a_y for user U_j. Here we use the 1-9 scale Paired comparison method to analyze the compared importance level of each attribute of the product that a user evaluates [19]. The rating matrix of an item is mainly acquired through user scoring, or acquired through user behavior analysis, or acquired with the approaches of Web semantic Suppose user U_j has rated several items and the rating matrix set is $AS = \{A(U_j, P_1), A(U_j, P_2), \ldots A(U_j, P_t)\}$, where $A(U_j, P_i)(i = 1, 2, \ldots t)$ is user U_j's rating matrix for product i (i.e., P_i). This paper applies the rating matrix set to establishing the user interest model. The specific steps are as follows.

1. Calculating the feature weight vector of each rating matrix, and then acquire the feature weight vector set

$VS = \{V_{U_j}^{P_1}(w_1, w_2, w_3, \ldots w_{size(A(U_j,P_1))}), V_{U_j}^{P_2}(w_1, w_2, w_3, \ldots w_{size(A(U_j,P_2))}) \ldots V_{U_j}^{P_t}(w_1, w_2, w_3, \ldots w_{size(A(U_j,P_t))})\}$.

Where $V_{U_j}^{P_i}(w_1, w_2, w_3, \ldots w_{size(A(U_j,P_i))})$ denotes the feature weight vector of the user rating matrix $A(U_j, P_i)(i = 1, 2, \ldots t)$ and $size(A(U_j, P_i))$ denotes the length of the feature weight vector.

2. According to the category of each attribute, calculate the user interest weights of the relavant attribute in the related resource category. Referring to the methods, we propose the following formula for calculating the degree of the user interest.

$$Va(U_j, a_y, n) = \frac{\sum_{k=1}^{n} A(U_j, P_k) \times V_{U_j}^{P_k}(w_y)}{n} \quad (1)$$

Where $Va(U_j, a_y, n)$ denotes the degree to which user U_j is interested in attribute a_y. n is the number of the items which has attribute a_y and user U_j have rated. $A(U_j, P_k) \times V_{U_j}^{P_k}(w_y)$ (k=1, 2, 3,…n) denotes the degree of user U_j's interest in attribute a_y of product P_k, which indicates how user U_j's preference on item P_k is mostly determined by attribute a_y.

3 Collaborative Filtering Recommendation Algorithm Based On Multi-attribute Group Decision-making

Firstly, we introduce the calculation of the value of user impact weight. This value is an important indicator to measure the degree of evaluation information consistency between a user and the others. The user matrix with higher group evaluation consistency will get higher weight. Vise versa. This paper adopts the concept of user rating similarity [6, 7]. We turn all the similar $n \times n$ user rating matrixes into $n^2 \times 1$ one dimensional vector. The user U_k judgment matrix A^k could be denoted as $V^k = \{\omega_{11}^k, \omega_{12}^k, \omega_{13}^k ... \omega_{1n}^k, \omega_{21}^k, \omega_{22}^k, \omega_{23}^k ... \omega_{2n}^k ... \omega_{n1}^k ...... \omega_{nn}^k\}$. Pearson similarity formula to calculate the rating matrix between the user U_k and the user U_l is show as follows:

$$Si(A^k, A^l) = \frac{\sum_{i=1}^{n^2} (V^k(i) - \overline{V^k}) \times (V^l(i) - \overline{V^l})}{\sqrt{\sum_{u=1}^{n^2} [V^k(i) - \overline{V^k}]^2} \times \sqrt{\sum_{u=1}^{n^2} [V^l(i) - \overline{V^l}]^2}} \tag{2}$$

$\overline{V^k}$ is the average value of all elements of user U_k rating matrix.

$$\overline{V^k} = \frac{\omega_{11}^k + \omega_{12}^k + ... + \omega_{nn}^k}{n^2}.$$

The similarity between user k and other users could be calculated as follows:

$$Si_k = \sum_{l=1, l \neq k}^{p} Si(A^k, A^l) / (p-1).$$

where p denotes the number of users. We propose a formula $D_k = 1 - Si_k$ as the approximate measure of variance, which indicates the deviation degree of evaluation matrix. The approximate influence weight of user U_k is shown as follows:

$$\theta_k = \frac{(1 - \max\{Si_l, l = 1, 2, ... p\})^2}{D_k^2} \tag{3}$$

After acquiring the similar user influence weight, we suppose the group integrated approximate evaluation matrix of p users is A^*, and the value of each element ω_{ij}^* in matrix A^* is as following:

$$\omega_{ij}^{*}=\sum_{k=1}^{p}\theta_k\times\omega_{ij}^{k}/\sum_{k=1}^{p}\theta_k \tag{4}$$

Matrix A^{*} is not a positive reciprocal matrix. Suppose *X* is a positive reciprocal matrix composed of x_{ij} . This paper uses the least square method to modify *X* and propose the following formula:

$$F(X)=\min\sum_{i=1}^{n}\sum_{j=1}^{n}(x_{ij}-\omega_{ij}^{*})^2$$
$$s.t\begin{cases} x_{ij}\times x_{ji}=1 \\ x_{ij}>0 \quad (i,j=1,2,\ldots n)\end{cases} \tag{5}$$

There are two important indicators in this article: compatibility and comprehensive compatibility. Their definitions are as follows:

Definition 1: Suppose *X* is the group user comprehensive evaluation matrix obtained by using the method of the least squares. Then the judgment matrix compatibility between user k and the other users is as follows:

$$S(A^k,X)=\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\omega_{ij}^{(k)}\times x_{ij}}{\max((\omega_{ij}^{(k)})^2,(x_{ij})^2)}}{n^2-\alpha} \tag{6}$$

Although paper [25] has defined expert judgment matrix compatibility in usual cases, it does not consider the incomplete value. Formula [6] is a modified approach to solve the problem. Firstly, the block that the user does not rate is processed and given the value 0. Then α is used to indicate the number of 0. The aim of this approach is to eliminate the influence of user judgment matrix on compatibility indicator.

Definition 2 : Suppose $A^{1'}$ 、 $A^{2'}\ldots A^{p'}$ are the compatibility correction of matrixes of *p* users' judgment matrixes. Then we get the comprehensive consistency indicator $\bar{S}$, as follows:

$$\bar{S}=\frac{\sum_{k=1}^{p}S(A,A^{k'})}{p} \tag{7}$$

Readers can refer to the simulation result of article [19]. When S(A,B)>= 0.8, the two evaluation matrixes is considered nearly compatible. When $\bar{S}\geq 0.8$, evaluation matrixes of all p similar users is considered compatible.

4 Experimentation

In order to validate the effectiveness of this algorithm, we build an experiment environment to execute our algorithm at current conditions. The environment is described as follows:

Ontology and the relevant methods are adopted to design and develop the movie information database. Jena 2.6.2 is applied to store the movie information in RDF format and ARQ-2.2 is used to manage the movie information. We have imported 300 movies which involve 10 categories. A semantic analysis of each movie is conducted to get key words and form the initial attribute set. Then the synonyms and the similar words in the initial set are combined. Take some topical words as the characteristic attributes and use them to represent these movies. Finally, 15 attributive categories and 282 concrete attributes are extracted. Then an online multi-attribute rating system based on the movie database and a collaborative filtering recommendation system based on group-decision making are designed and developed.

The concrete process that tests the algorithm is as follows:

1. Select four evaluated movies in which $G(u,p)$ is comparatively big and use them as the testify set. They respectively include 6,7,8 and 9 attributes. Then, use the target user evaluation matrixes which are further used as the real weight vectors to calculate the user interest vectors for each movie.

2. Based on the user-evaluated movies set (excluding the 4 movies in the test set), apply the methods in sections 2.3 and 2.4 to searching the most similar user set for the target user (i.e., the similar interest distributions).

Take Movie 1 with 6 attributes as an example. The real interest vectors are S=[3.7288, 2.7053, 1.9627, 0.4657, 0.3293, 0.3293]. The total score of this movie is 4.5 which indicates that the target user has a high preference to this movie. Moreover, the preference is mainly determined by the first three attributes. Totally, 9 similar users have evaluated this movie. Firstly, the traditional collaborative filtering algorithm is applied to obtaining the weighted average of the total score of this movie and gets the result 3.94. We are not sure whether the target users have interests in this movie. Thus, we need use the similar user evaluation matrixes to make judgments. The evaluation matrixes of six similar users are listed as follows:

$$A=\begin{bmatrix} 1 & 2 & 2 & 5 & 7 & 9 \\ 1/2 & 1 & 1 & 6 & 5 & 6 \\ 1/2 & 1 & 1 & 7 & 7 & 6 \\ 1/5 & 1/6 & 1/7 & 1 & 1 & 2 \\ 1/7 & 1/5 & 1/7 & 1 & 1 & 3 \\ 1/9 & 1/6 & 1/6 & 1/2 & 1/3 & 1 \end{bmatrix} \quad B=\begin{bmatrix} 1 & 2 & 3 & 3 & 7 & 7 \\ 1/2 & 1 & 0 & 4 & 7 & 6 \\ 1/3 & 0 & 1 & 0 & 4 & 5 \\ 1/3 & 1/4 & 0 & 1 & 1 & 2 \\ 1/7 & 1/7 & 1/4 & 1 & 1 & 3 \\ 1/7 & 1/6 & 1/5 & 1/2 & 1/3 & 1 \end{bmatrix} \quad C=\begin{bmatrix} 1 & 2 & 2 & 4 & 0 & 0 \\ 1/2 & 1 & 1 & 3 & 0 & 0 \\ 1/2 & 1 & 1 & 2 & 0 & 0 \\ 1/4 & 1/3 & 1/2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D=\begin{bmatrix} 1 & 2 & 3 & 4 & 6 & 8 \\ 1/2 & 1 & 2 & 2 & 4 & 6 \\ 1/3 & 1/2 & 1 & 1/2 & 4 & 7 \\ 1/4 & 1/2 & 2 & 1 & 3 & 2 \\ 1/6 & 1/4 & 1/4 & 1/3 & 1 & 2 \\ 1/8 & 1/6 & 1/7 & 1/2 & 1/2 & 1 \end{bmatrix} \quad E=\begin{bmatrix} 1 & 2 & 0 & 7 & 8 & 0 \\ 1/2 & 1 & 2 & 8 & 7 & 7 \\ 0 & 1/2 & 1 & 0 & 7 & 8 \\ 1/7 & 1 & 0 & 1 & 1 & 2 \\ 1/8 & 1/7 & 1/7 & 1 & 1 & 1 \\ 0 & 1/7 & 1/8 & 1/2 & 1 & 1 \end{bmatrix} \quad F=\begin{bmatrix} 1 & 2 & 4 & 0 & 7 & 0 \\ 1/2 & 1 & 2 & 6 & 7 & 7 \\ 1/4 & 1/2 & 1 & 0 & 2 & 2 \\ 0 & 1/6 & 0 & 1 & 0 & 2 \\ 1/7 & 1/7 & 1/2 & 0 & 1 & 1 \\ 0 & 1/7 & 1/2 & 1/2 & 1 & 1 \end{bmatrix}$$

(3) Use the following four algorithms to calculate the score of the four movies and make comparisons on the deviations of the real weight vectors of the target users. The result is listed as follows:

Table 1 the comparison between algorithms

	Movie 1 (6 order)	Movie 2 (7 order)	Movie 3 (8 order)	Movie 4 (9 order)
Arithmetic weighted average method	0.1589	0.0564	0.1985	0.1132
Logarithmic least squares method	0.1054	0.0534	0.1398	0.0831
Compatibility correction algorithm	0.0877	0.0556	0.1042	0.0687
Our algorithm	0.0780	0.0543	0.0885	0.0683

As shown in Table 1, when the scores of a part of similar users have a large deviation from those of the other users, the algorithm proposed in this paper can solve the problem of early convergence better than the other algorithms and obtain an accurate result. The core of our algorithm is the revised values of the comprehensive evaluation matrix determined by the majority of users. On this basis, the highly deviated evaluation values are revised. The result of seven order matrix experiment shows that the deviations of the result of any algorithms are not notable when all the similar users have unanimous evaluation matrixes,. The result of nine order matrix experiment shows that the result of the proposed algorithm is similar to that of compatibility correction algorithm when all the similar users have unanimous evaluation matrixes, while still have some incomplete values, and is better than the other two algorithms obviously. When there are 5 similar users and six order evaluation matrix is executed/ implemented with our algorithm, the change tendencies of the main indicators are shown in Figure 1:

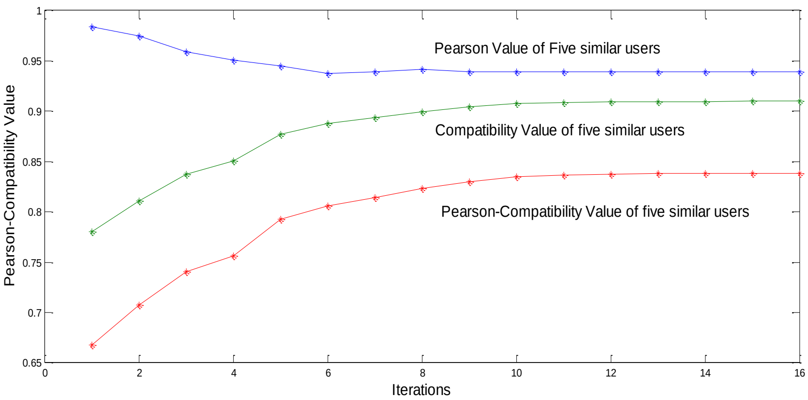

Fig.1: The main indicators change in our example

(4) Take movie one with the six order evaluation matrix as an example. The influence of the user number on the accuracy of recommendation results is examined. Suppose the user number is 3,5,7 and 9. The accuracy and the number of iterations are calculated with different means of permutation and combination. Part of the result is shown in Table 2:

Table 2: the comparison of different similar user numbers

User number	3	5	7	9
initial Indicator of Comprehensive consistency degree	0.5872	0.6890	0.6872	0.7081
Deviation of result	0.1680	0.1093	0.0828	0.0780
Number of iterations	12	16	28	37

When the user number increase, more deviation items are generated. Thus, the iterations of this algorithm raise. This test indicates that the effectiveness of this algorithm is highly related to the initial consistency degrees of all users and the number of users. In general, when the initial consistency degree is low and the similar user set is limited (e.g. there are 3 users), it is hard for the algorithm to dig out the common information among the users. Therefore, the result deviation is huge. However, when the number of similar users increases to a certain degree (e.g. the number is equal or bigger than 7), this algorithm still remains a good accuracy, even if the initial compatibility is low.

In the aspect of providing personalized services to the target users, this paper calculates the comprehensive evaluation weight vectors of each movie with the group-decision making model. Take movie one with 6 attributes as example. The comprehensive evaluation score of nine similar users is $G(u,p)$ =3.94. The comprehensive evaluation vectors are V= [4.0653 2.9492 1.7630 0.3972 0.3044 0.3607]. Each value of the weight vector represents the potential interest degree of the target user on the corresponding product attributes. Thus, the total score calculation formula is

$$TScore = G(u,p) \times \sum_{i=1}^{n} V_i \ / n \tag{10}$$

where *TScore* denotes the total score, *n* denotes the number of attributes, and V_i denotes the comprehensive evaluation value of the (*i*)th attribute of the product. The recommendation set can be fixed through the way of ranking or threshold setting. The total scores of the four movies is shown in Figure 2:

As shown in Figure 2, the movie one has the highest score with six attributes. The characters of this movie are analyzed as follows using the user interest model. Firstly, attribute V_i of weight vector V is normalized and generate vector $\overline{V} = [0.4131, 0.2997, 0.1792, 0.0404, 0.0309, 0.0367]$. The three attributes whose values are bigger than the average value 0.1666 are picked out. When the attribute value is bigger than 0.1666, the majority of users have evident preference on movie one. In the target user interest model, there are 124 attributes totally. The 3 attributes of movie one that are bigger than the average value are connotation, characteristic and

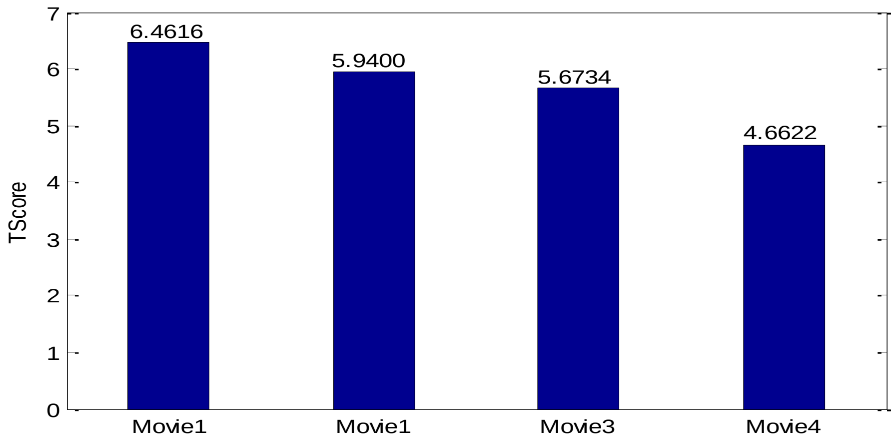

Fig.2 The TScore of four movies

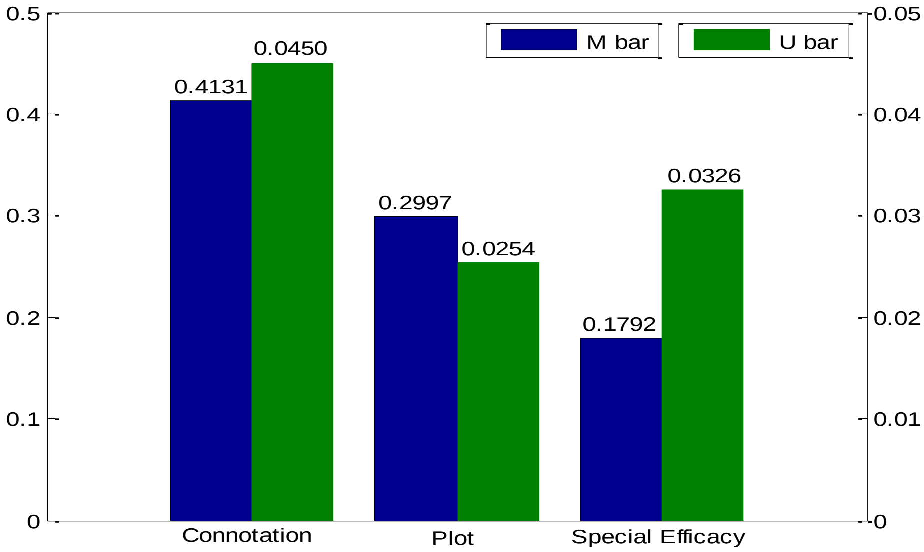

Fig. 3: Comparison analysis histogram of movie one with target user interest.

special efficiency are still larger than the average value (1/124=0.0081) among 124 target user attribute preferences. This result indicates that target user has evident preference to these 3 attributes and the popularity of this movie is mainly determined by these attributes. Therefore, we could introduce movie one to target user and provide the reasons why this movie is introduced. We also could use semantic analysis technique to describe each attribute in detail and provide more personal service to target user. The comparison analysis histogram is shown in Fig. 3.

5 Conclusion

Currently, the collaborative filtering personal recommended algorithm lesser consider the multi-attribute problem. We take the method which is based on the group-decision making, then build an improved Pearson-Compatibility algorithm and apply it into the collaborative filtering recommend field. And we also build a virtual recommend environment and testify the effectiveness and feasibility of this algorithm. The collaborative filtering personal recommended algorithm which is based on group-decision making have some advantages as following:

Firstly, we could find a more suited similar users set for the target user. Through field subdivision based on field attributes, we could get a more accurate target user model. Take the model as a foundation, we could find users who have similar interest distribution with target user and build the similar users set.

Secondly, our algorithm could provide more accurate and personal recommend service to our target user. The traditional collaborative filtering method merely could recommend a result set to target user, but could not provide analysis service. Our method overcome this weakness, making a information integration to know what are mainly factors determining the user preference, so that we could handle the user need more accurate.

Thirdly, we consider evaluation deviation between the similar users and revised the user evaluation. Compared with the traditional method of weighted mean, we use group-decision making method to calculate the comprehensive evaluation score. We believe deleting the deviation item and revising the evaluation matrix could make the result have a better fitting effect. We also applied the collaborative filtering method which is based on Pearson-Compatibility to the personal recommended field. The experiment result shown that this algorithm is stable when facing the deviation items and could find the common preference information between similar users.

If we want to apply this algorithm to real business environment, we need a new user evaluation model as a foundation. For there are little related research currently, the next step is to apply internet technique to build user online evaluation system. After collecting lots of user data, we could research this model further to testify its effective.

References

1. Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowl. and Data Eng., 17(6):734–749, 2005.
2. Xiaoyuan Su and Taghi M. Khoshgoftaar ,A survey of collaborative filtering techniques. Advances in Artificial Intelligence, Volume 2009 (2009),

3. D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," Communications of ACM, vol. 35, no. 12, pp. 61–70, 1992.
4. P. Resnick and H. R. Varian, Recommender systems, Communications of the ACM, vol. 40, no. 3, pp. 56–58, 1997.
5. K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, Eigentaste: a constant time collaborative filtering algorithm, Information Retrieval, vol. 4, no. 2, pp. 133–151, 2001.
6. Luis M. de Campos, Juan M. Fernández-Luna ,Juan F. Huete ,Miguel A. Rueda-MoralesCombining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks, International Journal of Approximate Reasoning,Volume 51, Issue 7, September 2010, Pages 785–799