MATEC Web of Conferences **169**, 01003 (2018) https://doi.org/10.1051/matecconf/201816901003
IMETI 2017

IPTV program recommendation based on combination strate gies

Hao Li, Huan Xia, Yan kang and Mohammad Nashir Uddin[a]

School of Software, Yunnan University, Kunming, China

Abstract. As a new interactive service technology, IPTV has been extensively studying in the field of TV pro-gram recommendation, but the sparse of the user-program rating matrix and the cold-start problem is a bottleneck that the program recommended accurately. In this paper, a flexible combination of two recommendation strategies proposed, which explored the sparse and cold-start problem as well as the issue of user interest change over time. This paper achieved content-based filtering section and collaborative filtering section according to the two combination strategies, which effectively solved the cold-start program and over the sparse problem and the problem of users interest change over time. The experimental results showed that this combinational recommendation system in optimal parameters compared by using any one of two combination strategies or not using any combination strategy at all, and the reducing range of MAE is [2.7%,3%].The increasing range of precision and recall is [13.8%95.5%] and [0,97.8%], respectively. The experiment showed better results when using combinational recommendation system in optimal parameters than using each combination strategies individually or not using any combination strategy.

1 Introduction

Currently, IPTV, DTV, and the Internet have been using widely, and the number of TV programs is rapidly growing. It usually takes a long time for us to find out what we prefer from hundreds of TV programs. Previously there were not too many programs due to various limitations and restrictions on TV channels. Then only few TV programs are selectable for the audience. Also, EPG (Electronic Program Guide) technology relieves the information overload of the TV program. With the number of TV programs increasing, EPG has become more and more inadequate. Therefore, some recommendation systems based on EPG have proposed 0, 0.

IPTV (Internet Protocol Television) is also known as network TV. It takes home TV as the primary terminal equipment through using the infrastructure of a broadband cable television network, to provide a variety of digital media services including television programs. The provider involved in deploying IPTV services that range from cable and satellite TV carriers to large telephone companies and private network operators.

With the rapid development of the Internet, programs and movies (from now on collectively referred to as TV programs) emerge in large numbers. In this background, if the user cannot quickly find their favorite programs, the ratings or demand rate will decline, and users will gradually reduce

[a] Corresponding author : mun2003dhk@gmail.com

their reliance on IPTV, which eventually leads to decline in service quality and some users. Therefore, how to quickly find the user's favorite TV programs has great significance for the healthy development of IPTV. Recommendation system as an information filtering technology can efficiently solve above problems. However, IPTV unlike conventional television and its characteristics presented as follows.

Interactivity: differently from traditional TV, where the communication is unidirectional, the two-way capabilities of IPTV systems allow the user to interact with the system. In particular by allowing IPTV systems to explicitly or implicitly collect user preferences.

Accuracy: IPTV to deliver multimedia content to the user IP address, then the service provided by IPTV service provider can be accurate to individuals.

Multi-Media: IPTV services include scheduled television programs and video on demand (VoD) contents.

Freedom: Users can choose which TV program they want to watch according to their wishes at any time.

As above characteristics of IPTV shown, first of all, the accuracy of IPTV provides the implementation condition for the recommendation system. Secondly, the IPTV provides the data support for the recommendation system, which guarantees the quality of the recommendation. Finally, IPTV's multimedia and the freedom of the IPTV recommendation system provide the necessary support. Therefore, the recommendation system is integrated into IPTV infrastructure, and the recommended results are calculated by using the explicit or implicit data collected by the IP network. The guidance will give users faster and more exciting way to select TV programs and will positively promote the healthy development of IPTV business.

IPTV recommendation system is still facing many challenges. In the field of IPTV, due to the increasing number of users and the difference of the users' choice, so that the difference of user's rating is enormous. At the same time, the problem of user-rating matrix sparsity caused by lacking user ratings because of the substantial increasing of challenges and the cold-start problem caused by newly added items or users have a significant influence on the quality of the recommendation result.

The recommendation algorithm currently used in IPTV is collaborative filtering algorithm and content-based recommendation algorithm. Collaborative filtering algorithm is divided into user-based collaborative filtering algorithm 0 and item-based collaborative filtering algorithm [8]. User-based collaborative filtering algorithm is based on the assumption that if the users' rating for the items is similar, their rating will be similar for other items as well. Item-based collaborative filtering algorithm holds that users' rating for the items has similarity 0.

Content-based recommendation tried to recommend items that are similar to the items user liked in the past, through matching user interest model and contents of an item. According to the characteristics of content-based filtering, we can know it does not have a sparse problem and item cold-start problem. But its recommendation results are too customized to simply find items which are different from the items the user liked in the past 0, 0.

Relevant researchers propose some other recommended algorithm in IPTV based on collaborative filtering algorithm and content-based filtering algorithm. Such as LSA (latent semantic analysis) content-based algorithm, SVD content-based algorithm, weighted combination algorithm for based filtering and collaborative filtering algorithm, switch combination algorithm for content-based filtering and collaborative filtering algorithm. LSA content-based algorithm increases additional latent semantic analysis for the plot of different TV programs. SVD content-based algorithm takes advantage of singular value decomposition to decompose user-program metadata matrix W, whose elements w_{ci} represent the relevance of characteristic (metadata) c for item i. These two kinds of improved algorithm based on the content-based recommendation can improve the quality of recommendation, but there is still no solution to the fundamental problem, user cold-start problem, over-customization problem, and the problem of user interest change over time. The following two combination algorithms, just only put together two kinds of algorithms, is not obvious to the quality of the recommendation results.

This paper combined the advantages and disadvantages of collaborative filtering, content-based filtering, SVD, combination algorithm to solve above problems. The original data is separated into two pieces. The data which is favorable to the collaborative filtering algorithm is handed to collaborative filtering algorithm, and then the content-based filtering algorithm processes the remaining data. The two pieces of recommendation results were integrated into the final recommendation result. The result can restrain the non-ideal problem that comes from sparsity problem of collaborative filtering algorithm for recommendation results. However, when data volumes and data sparsity are particularly significant, the data handed to collaborative filtering algorithm may be still sparse. SVD has been performing well for the sparse problem. Nevertheless, the paper [12] proves that SVD instead is not as good as collaborative filtering algorithms when the sparse degree is still gigantic. Its main reason is that the data integrity is required in the simple SVD algorithm, and the general approach is to use 0 to fill up, but this simple way of filling does not accurately represent the user's particular rating. That leads to the SVD calculation error is enormous. Therefore, can fill the predicted data outputting from the collaborative filtering to the missing data in SVD to reduce the error caused by the SVD in the missing data occupied by 0? If the hypothesis (hypothesis 1) is true, then separation of data improves collaborative filtering once again enhances the quality of collaborative filtering, the quality of SVD. Therefore, the process of collaborative filtering is transferred to the process of SVD. Secondly, [17] pointed out that the SVD may not be as collaborative filtering algorithm to behave well when the data is very dense. Then, can the compact data calculated by SVD take advantage of again collaborative filtering algorithm to (or "intending to") further enhance the quality of recommendation? If the hypothesis (hypothesis 2) is true, the SVD process shifts to the collaborative filtering process. Finally, the data handed to content-based filtering will face to the massive sparsity, but content-based filtering is not sensitive for the sparsity of data, and the over-customization problem also will be restrained considerably in collaborative filtering. Meanwhile, there are two issues. Issue 1: how to build user interest model in the section of content-based filtering? Issue 2: how to update user interest model?

This paper carried out research based on the above idea and some of the problems. This System was named ICR (IPTV Combined Recommender) according to two combination recommendation strategies proposed by this paper. ICR includes the section of collaborative filtering and the section of content-based filtering. The elaboration of two kinds of combination strategies was placed in Chapter III.

2 Related work

With a broad range of information overloads resulting in the need for information filtering. In the field of television, here comes the emergence of personal TV program recommendation system: Based on Metadata TV-anytime recommendation system [13], PDPR [14], etc.

With the broad use of cloud computing technology, PDPR has adopted the cloud computing technology in the field of TV recommendation. PDPR based on cloud computing technology to collect and analyze the user's view mode (such as the view mode of mobile phones or personal computers, etc.) to achieve the purpose of recommending digital TV programs. For example, PDPR can analysis the user's interest through analyzing the data generated from communications services, mobile phone, and personal computer. The recommendation method determining the weight value of the time characteristic by analyzing the proportion of user's viewing time in the total time of the program is still based on the content filtering recommendation.

There is another disadvantage to this method besides the problem proposed in the content-based recommendation. It is quite uncertain to determine the degree of user's interest only depending on the proportion of user's viewing. The users, for example, go to do other things with the TV is on. Except that, the user left due to some other things with seeing a little time for the program. The system will regard the user's action as a moderate preference for this program. However, the truth may be that the user fond of the program very much. PDPR does not provide adequate solutions for the above problems.

The paper 0 proposed a method of content recommendation based on context awareness to provide users a more advanced IPTV services. Its background comes from users, devices, services, networks, and each setting characteristic is assigned a corresponding weight value. The user is divided into N group, and the similarity between each group and the content is calculated by measuring the average weight of its context characteristic. At last, the highest N content is recommended to the user group.

The paper [4] proposed three algorithms for IPTV, which is based on the substance of the latent semantic analysis algorithm, the SVD algorithm, and the content based SVD algorithm. LSA content-based algorithm increases additional latent semantic analysis for the plot of different TV programs. SVD content-based algorithm takes advantage of singular value decomposition to decompose user-program metadata matrix W, whose elements w_{ci} represent the relevance of characteristic (metadata) c for item i. Firstly, these three algorithms are not useful in combination, but for a particular class of problems (such as cold start) in isolation. These algorithms cannot solve many problems simultaneously, and the recommendation quality is not ideal when the simple SVD algorithm is in the calculation of heavy sparse. To compare with the method proposed by this paper, it also solves the problem of cold-start, sparsity, and the problems of recommendation results are too customized. Secondly, LSA content-based algorithm of paper [4] does not consider the factor of user interest, which will lead to the deviation of the user interest model and the real situation. This paper proposed a new method to update the user profile to minimize the difference from the actual situation. Finally, differently from the SVD algorithm the paper 0 propose, in the first combination strategy this paper's propose to combine SVD and collaborative filtering algorithm. SVD is not a simple decomposition of the original matrix, but the characteristic increment combination of collaborative filtering and SVD is used to solve the influence of the matrix over sparse for the accuracy of the decomposition.

The paper 0 proposed a prediction method of item keywords in the content-based recommendation. This method uses the window algorithm to update the user profile, but this paper is different from the user profile update rule of 0 where the original weight is just multiplied by or divided by a fixed value, two. The value does not take into account the impact of the user's rating, the number of keywords in the target program, and the occurrence number of keywords on the weight change, which leads to the increase of the calculation error.

Some other methods are also proposed, for example, the paper [5, 6] regard time and external burst news event as a contextual factor to improve the quality of the recommendation algorithm.

3 Combination strategy and architecture

Collaborative filtering item based have been proved that it has an advantage over user-based collaborative filtering [18] in the area of interest is fixed (such as books, movies, e-commerce, TV shows, etc.), so this paper adopts a collaborative filtering algorithm for item-based collaborative filtering algorithm in this paper.

Based on the idea proposed in Chapter 1, the issue 1 is how to build a user interest model, and that needs to solve first. The use of program content data comes from the EPG content or other content providers in ICR systems proposed in this paper. ICR provides a user interface, where users can register. The registration requires the user to provide specific interest preference, and the preference will be stored in the user profile database in the form of user profile (user profile is a structured user interest model shown). The user interface also allows a user to provide explicit rating through interaction with IPTV for a particular TV program, and the score data will be stored in user rating database. After each time, the user rate a program, a user profile will be updated according to the content provided by the content provider.

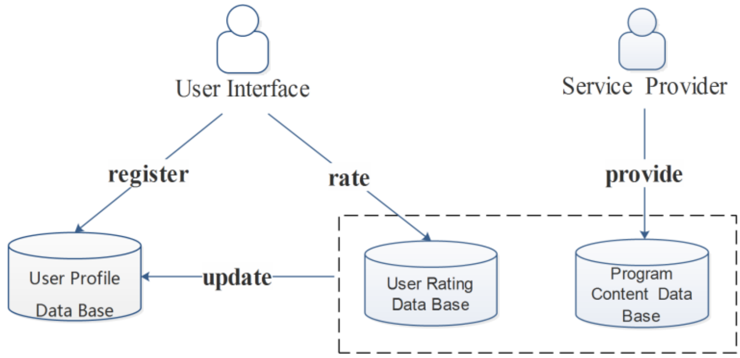

Figure 1. Business Process.

These data will be used as input data of recommendation algorithm. It's business processes as shown in Fig.1.

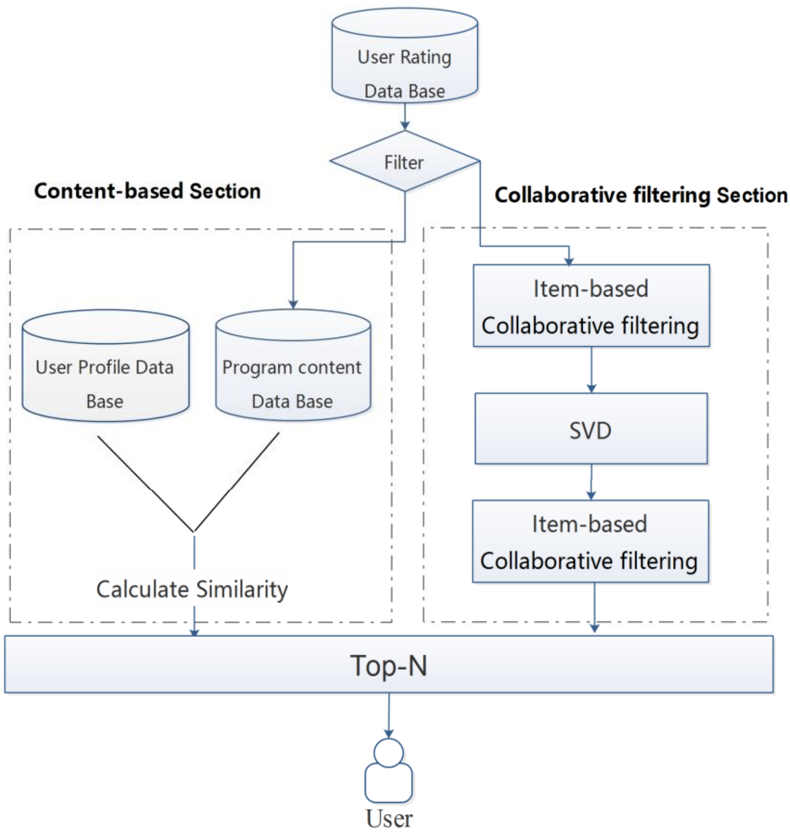

Figure 2. Architecture

In front of Chapter Ⅰ idea, the data is separated, which is the second recommendation strategy that proposed in this paper. The data filter section as shown in Fig. 2 implement the second recommendation strategy. Its role is to determine which recommendation section the data belongs. When the number of user's rating of a program accounted for a proportion less, the parameter m in the total amount of users. This program will be recommended for content-based filtering section, otherwise through collaborative filtering section. The basis of the judgment rule is: In the related similarity measure method if the user set which rated the item I is represented by U_I. The user intersection U_{ab} of two users set (U_a *and* U_b) which respectively measured item a and item b need calculating.

$$U_{ab} = U_a \cap U_b \tag{1}$$

The similarity between item a and b is calculated by using the method of correlation similarity measure on the set U_{ab}. According to the idea of item-based collaborative filtering, only the rating is similar in more number of a user; then we have a high determinacy for similarity between the items. However, in the case of user - program rating data extremely sparse, the set of elements in U_{ab} is microscopic. Even if the similarity of the two items calculated in such a situation is very high, we cannot be sure that the similarity between them is very high. When the number of user's rating of a program accounted for a proportion less than parameter m in the total amount of users, the user value of the co-rating between this program and other programs will be a better chance of little. The following experiments will optimize the parameter m:

3.1 Content-based filtering section

Unlike Web-based IPTV field, such as the content of news sites which are text type can get desired content by keywords and information classification, etc., and also obtain information through simple mouse and keyboard actions.

IPTV content for video content and video content is not very efficient for all video content including every frame of the picture. In this context, at present, the content data (actor, type, Director, etc.) of TV program is used to describe TV shows.

The online registration of ICR system requires an explicit user preference, and all choices are divided into three types of characteristic values, respectively actor, and director.

Each type of features is allowed to have one or more keywords, and each keyword has the light to represent the relative importance of keywords, as well as has the keyword occurrence in the program rated by the current user. At the time of registration for each keyword of each type of feature, weight is given initial value.

3.1.1 User profile update method

Then solve the second problem (issue 2), that is, after building user interests model, to adapt to the users ' interests and to improve quality content-based recommendation, how to update a user profile? A user profile update method is proposed in this paper. If each type of domain of the three types of characteristics is regarded as related k-keywords, then each type can be described by the k-keywords. For example, type (action, Sci-Fi, comedy, horror), Actor (Jackie Chan, Bruce Lee, Jet Li, etc.). A user profile is defined as triples {key, w, times}, where the key represents a feature value keyword, w accounts for the equal weights, times represent the current occurrences of the keyword. Assume a user profile of the actor class shape, actor (Jackie Chan, 1,2), which accounts for a weight value of particular actor "Jackie" is 1, occurrences are 2. When a TV program which a user has rated, the show's actors for "Jackie" and "Andy Lau" is matching the program associated with the user profile type keywords.

$$w_{new} = w_{pre} * \frac{rating}{\sum w_i x_i} * f(x) \tag{2}$$

If the keyword exists in the user profile is in the program, then using the formula (2) updates the keyword weight, otherwise, add the keywords and assign initial values.

If a user likes a higher rating program or specific keywords to find a program, then the corresponding keyword weight would increase more or less. In this paper proposed a keywords weight update formula based on this idea.

w_{new} represents the target keyword weights updated. w_{pre} represents the weight of the target keywords before the update. $rating$ represents the rating the user rated on the current program. $\sum w_i x_i$ represents the sum of keywords weight extracted from this program content, which i represent i-th keyword in this type of feature. $f(x)$ is one of the independent variable x (x accounts for the target keyword occurrences) of the nonlinear function. Is mainly based on the fact that the more occurrences of targeted keyword? And the corresponding keyword weight to increase exponentially. The αβ values may be determined according to specific application needs, f (x) is defined as:

$$f(x) = \alpha e^{\beta x} \tag{3}$$

3.2 Content-based filtering recommendation algorithm

Content-based Algorithm is named CB Algorithm, CB Algorithm is shown as following:

Algorithm 1: Contend-based
Input: user ID, item ID
Output: predicted rating matrix

Process:

a) Building vector space model: The user profile gotten from user profile according to the user ID and program content information got from program content database according to the program ID is constructed as a vector space model.
b) Calculation of the similarity between the two vector space model: Assume p, u, on behalf of the user vector and program vector, respectively. i represents i-th element in the vector.

$$sim(u,p) = \frac{\sum_i \left(u(i) \times p(i)\right)}{\sqrt{\sum_i u(i)^2} \times \sqrt{\sum_i p(i)^2}} \tag{4}$$

Formula (4) shows how to calculate the cosine similarity between program vector and user model vector.

The recommendation system will calculate the similarity between each programming vector and target user vector. And the similarity of the highest N program as the recommendation results of a content-based section.

A. Collaborative filtering Section

Hypothesis 1 presented in Chapter Ⅰ is that it be able to fill the predicted data from the collaborative filtering to the SVD part of missing data to reduce the error caused by the SVD in the missing data occupied by 0. Open GroupLens movie data set is used, the question was verified, the similar experiment was arranged in Chapter IV. Experiments show that this hypothesis is valid. Then the reasoning is in hypothesis 2, that is, the dense data calculated by SVD using collaborative filtering algorithm can interact to (or "intending to") further enhance the quality of recommendation. The paper [17] pointed out the SVD may not be as collaborative filtering algorithm to behave well when the data is very dense. Item-based collaborative filtering algorithm needs to calculate the similarity between the items. If a rating comes from two items never being the same user, which will lead to the similarity between the two items cannot be calculated. Therefore, the quality of recommendation results will be increased, if the density data derived from the SVD take advantage of collaborative filtering again. The second hypothesis was also established. So far, the feasibility of this idea was entirely justified.

The first combination strategy is proposed in collaborative filtering section of this paper. First of all, a user-program rating matrix will be created for those programs handed to collaborative filtering section. Secondly, the original rating matrix will be filled based on item-based collaborative filtering. Then, to put the user-program rating matrix filled as an input of singular value decomposition, and SVD will fill original user-program rating matrix again, and get a user-program rating matrix without missing value. Thirdly, the user-program rating matrix without losing value will be used item-based collaborative filter again, and the missing item of original will be predicted finally, regarding N programs with highest ratings as recommendation results of the collaborative filter section. Using algorithm among this process is named CF-SVD-CF.

Algorithm 2: CF-SVD-CF
Input: original rating matrix
Output: predicted rating matrix

Process:

a) Item-based Collaborative Filtering Predictions：original rating matrix R is defined to be a $m \times n$ matrix, which represents the rating of m users on the n items. Using the method of item-based collaborative filtering and measure method of the modified cosine similarity to calculate the similarity between items. The calculating formula is as follows:

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \overline{R_u})(R_{u,j} - \overline{R_u})}{\sqrt{\sum_{u \in U}(R_{u,i} - \overline{R_u})^2}\sqrt{\sum_{u \in U}(R_{u,j} - \overline{R_u})^2}} \tag{5}$$

In the formula (5), $\overline{R_u}$ represents an average of all ratings the target user u has rated, $R_{u,i}$ accounts for the rating the user u has rated on item i in the original matrix, U represent user set. To find the target, user u to the program j rating prediction, the formula is as follows:

$$Pre_{u,j} = \frac{\sum_{i \in ratedItem(u)} sim(i,j) \times R_{u,i}}{\sum_{i \in ratedItem(u)} sim(i,j)} \tag{6}$$

In formula (6) ratedItem (u) represent a set of the user u has rated. $sim(i,j)$ represent the similarity between program i and program j. Finally, the matrix R_{pre} is obtained by filing the prediction rating into original matrix R.

b) Standardized R_{pre}: If a rating comes from two programs never being the same user, the similarity between the items will not be calculated. So R_{pre} may not be filled entirely. If not, first calculating the average value $\overline{R_j}$ of each column in the R_{pre}, then fill the average column value $\overline{R_j}$ into the vacancy in the current column, and each column of the ratings of all items minus the target program where the average row value of $\overline{R_i}$ to get the R_{norm} matrix of standardized. The reason of standardized treatment is there is some different influence for calculating similarity in users who rated different number of items

c) Calculate for SVD:S, U, V, is obtained by using the singular value decomposition method for R_{norm}. Their sizes are $\times m\ m \times n$, $n \times n$. The relationship between them is $R_{norm} = U \times S \times V^T$.

d) Dimension Reduction: A $k \times k$ matrix S_k is obtained by preserving k-diagonal element for matrix S. Matrix U, and matrix V are also reduced to $U_k$and V_k accordingly. Their sizes are $m \times k$ and $n \times k$. The experiment will optimize the parameter of K.

e) Users and programs in k-dimensional space said: m a user in a k-dimensional feature space representation for K U * tick (s k) and n a program in a k-dimensional feature space representation for tick (s k) x (v k) ^ t.

f) SVD prediction rating: according to the [28], the target user u in the program I on the prediction rating is calculated:

$$Pre_{u,i} = \overline{R_u} + U_k\sqrt{S_k}(u) \times \sqrt{S_k}V_k^T(i) \tag{7}$$

Which $\overline{R_u}$ the said target user u all has hit rating project rating average, $\sqrt{S_k}V_k^T(i)$ on behalf of column i of the matrix $\sqrt{S_k}V_k^T$. $U_k\sqrt{S_k}(u)$ on behalf of the row u in the matrix $U_k\sqrt{S_k}$. A rating matrix R_{filled} without missing values is obtained by filling the predicting to the original rating matrix R.

Matrix R_{filled} is just a matrix obtained by using SVD method to predict the predicted matrix, which is not based on the method of collaborative filtering.

g) Similarity calculation: Using the modified cosine similarity measure method to calculate, the similarity between program I and program J in matrix R_{filled}, the formula is as follows:

$$sim(i,j) = \frac{\sum_{u\in U}(R_{u,i}-\overline{R_u})(R_{u,j}-\overline{R_u})}{\sqrt{\sum_{u\in U}(R_{u,i}-\overline{R_u})^2}\sqrt{\sum_{u\in U}(R_{u,j}-\overline{R_u})^2}} \tag{8}$$

$\overline{R_u}$ represents an average of all ratings the user u has rated $R_{u,i}$ represents the rating user u has rated on item i, U represents the set of users.

h) Final prediction rating of the original matrix: We conclude with prediction generation, achieved by the following weighted sum:

$$Pre_{u,j} = \frac{\sum_{i\in ratedItem(u)} sim(i,j) \times R_{u,i}}{\sum_{i\in ratedItem(u)} sim(i,j)} \tag{9}$$

i) The prediction calculates for user u on the item j. $ratedItem(u)$ represents the set of items user u has rated. $R_{u,i}$ represents the rating user u has rated on item i. $Sim(i,j)$ represents the similarity between item i and item j.

4 Evaluation

Evaluation is a core aspect of recommender systems design and deployment. The measurement of the quality of the assessment rating is divided into two categories: coverage rate and accuracy measure while the efficiency test includes two kinds of methods: statistical precision measurement method and decision support accuracy measurement method. MAE (Mean Absolute Error) is one of the commonly used recommendation quality measurement methods in statistical precision measurement method. MAE is predicted by calculating the deviation between the prediction and the actual user's rating. MAE is smaller, and the prediction for the user is more accurate [16], and the MAE formula is as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|P_i - T_i| \tag{10}$$

In the formula (9), N is a set of all the missing rating items. P_i represents the prediction for the user on target item. T_i represents the user's real rating on the target item. MAE is smaller, and the prediction for the user is more accurate. But MAE in this system is not related to that data which is handed to content-based section because the content-based section does not predict the user's rating. So the MAE values are calculated only for that part of the data in the collaborative filtering section.

Also, the recommendation quality is usually evaluated by recall and precision in traditional information retrieval field. The two measurement methods are still suitable for TV program recommendation system. But it is needed to modify the formula to obtain a more accurate measurement, $Set_{preferred}$ represents the set of users' preference, which involves the programs where the rating is equal or greater than the sum of all ratings the target user has rated and the standard error of all ratings the target user has rated. //Assumes that $item$ is one of the programs in $Set_{preferred}$; P_{ave} is the average of all items rated. P_{stdDev} is the standard deviation of all items rated. Then, the rating, $item_{rating}$, of $item$ must meet following conditions:

$$\text{If } item \in Set_{preferred} \tag{11}$$

$$\text{Then } item_{rating} \geq P_{ave} + P_{stdDev} \tag{12}$$

$Set_{recommend}$ represents the recommendation result set which is generated by the recommendation system. Recall and precision can be defined as follows:

$$recall = \frac{Set_{preferred} \cap Set_{recommend}}{Set_{preferred}} \tag{13}$$

$$recision = \frac{Set_{preferred} \cap Set_{recommend}}{Set_{recommend}} \tag{14}$$

When using the first recommendation strategy, due to the addition of content- based recommendation results, so $Set_{preferred}$ must also contain content-based recommendation result set.

B. Data Set

In the following experiments, this paper uses MAE, recall, precision as a measure of the quality of the system to assess the quality of the system. Adopting the public GroupLens movie data set, and using web crawler program to collect all the video content data for the experiments. GroupLens data set contains 1000209 rating data, from 6040 users in 3900 films on the rating. The value of the rating range is from 1 (dislike) to 5 (like). In this paper, the original data is divided into training set and test set, the training set is 80%, and the test set is 20%.

C. The experiment and analysis

In this paper, the collaborative filtering and content-based recommendation are implemented, and the quality of each recommendation strategy is evaluated through the switch or combination of the two sections. The experimental process of the development of programming language for the Java, the database is Mysql. In the collaborative filtering section, the implementation of item-based collaborative filtering algorithm and the SVD algorithm make the use of rewriting and combination of the algorithm in the open source project. Regarding content-based implementation, firstly, the collection of the film content data is used to establish program content table in [movieId, Mysql, genre, actors, director] on behalf of each program Id, type, actor and director fields. Secondly, the establishment of the user profile table to [userId, genre, actor, director] on behalf of the field of each user's Id, types and their corresponding weights, actors and their corresponding weights, directors and their corresponding weights, and using the training set to update the table. Finally, the use of the table and the programs handed to content-based filtering section to match and calculate, so to get the user for a program of similarity.

As mentioned in Chapter IV, the parameter m represents the proportion of the number of rating rated by the target user in the total user rating for a program. The parameter K accounts for the reservation count of singular value of the singular matrix S. First of all; the first implementation is carried out to prove the correctness of the first hypothesis (hypothesis one).

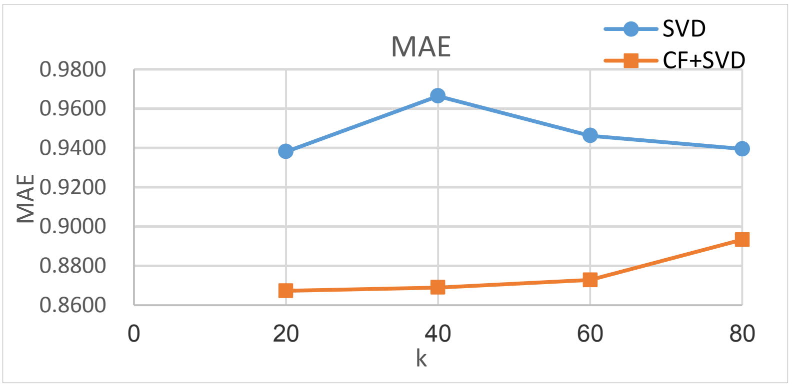

Figure 3. MAE of SVD and CF+SVD.

Fig.3 shows in the simple SVD algorithm with the parameter k={20, 40, 60,80} MAE are higher than the algorithm where collaborative filtering is used first to fill the original matrix. And then use SVD to predict the missing value of the original matrix. A conclusion, the first hypothesis is upright, can be drawn. That is, it can fill the predicted data outputting from the collaborative filtering to the missing data in SVD to reduce the error caused by the SVD in the missing data occupied by zero.

All of the following experiments were used to evaluate the recommendation quality of two kinds of combination recommendation strategies in different parameter m and K, respectively. To determine the optimal parameters m and k through experiments and observe the value of the corresponding parameters of MAE, Precision, and Recall.

The experimental results are shown in Fig. 4, Fig. 5, Fig. 6 with two combination recommendation strategies. The abscissa represents the value of k,50 as a unit. Parameter k is set to {5, 20, 40, 60, 80, 100, 150, 250}. Ordinate respectively represents MAE, Precision, Recall with the value of the corresponding k. Each polyline corresponds to an absolute value of m, m is set to {0.01, 0.005, 0.007}.

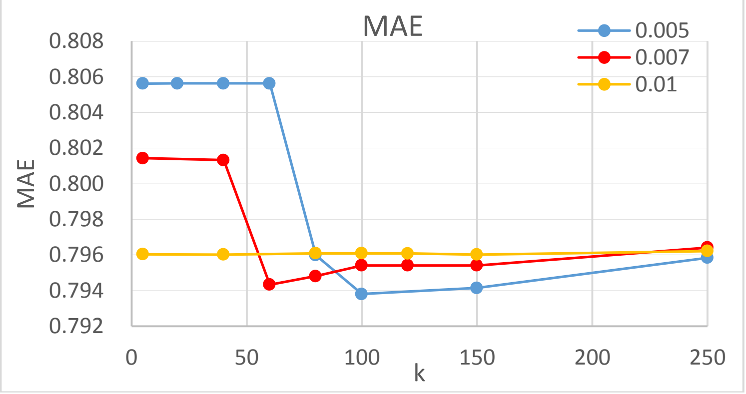

Figure 4. MAE with two recommendation strategies.

Fig.4 shows that, firstly, when m is 0.005, 0.007, and k is small, the MAE values decreased very slowly. But with increasing value of k, MAE dropped rapidly, and when k=60(m=0.007), k=100(m=0.005), the results reach the optimum. After that, with the increase of k, the results gradually become awful. Secondly, when m=0.01, MAE has maintained a gradual change, the minimum value is only 0.79602. Thirdly, when m=0.005, average recommendation quality is better than the other two.

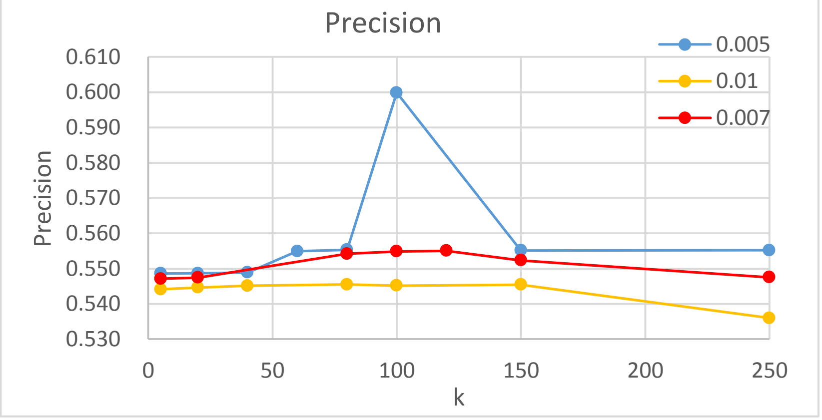

Figure 5. Precision with two recommendation strategies.

Fig 5. indicate that when m value of 0.005, 0.007, 0.01, with the increase of k, precision has small fluctuation. But when m=0.005, k∈ [80, 150], precision increases rapidly and k=100 reach highest value 0.599884.

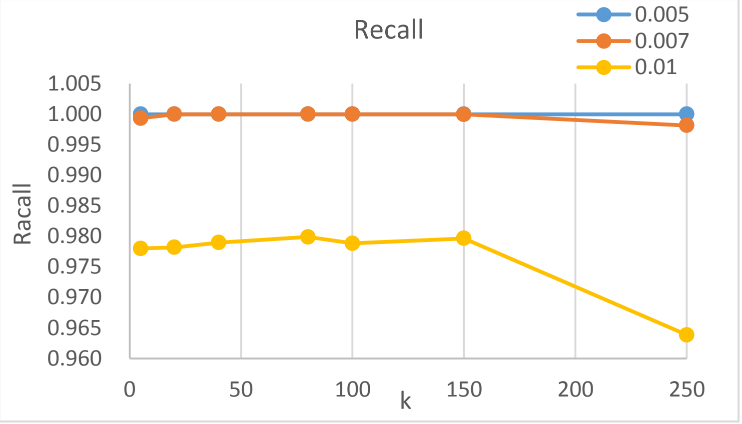

Figure 6. Recall with two recommendation strategies.

In Fig. 6, when m=0.005, Recall remained at 1.0, m=0.007, Recall has a gradual fluctuation about 0.999. But when m=0.01, and Recall rather dramatically reduce, the highest value is only 0.97988. When k ∈ [5, 150], the fluctuation is small, but when k increases to more than 150, Recall value dropped sharply, and k=250 reach the lowest value 0.96385.

This paper retained the second recommendation strategy to evaluate recommendation quality of the first recommendation strategy and canceled the first recommendation strategy. Parameter k takes value collection for {5, 20, 40, 60, 80, 100, 150, 50}, and an experiment is proceeded in same training set and test set. The first recommendation strategy involves the content-based recommendation to test MAE values, so MAE of two recommendation strategies is same as only the second recommendation strategy. Then the group experiment only tested the precision, recall. The experimental results as shown in Fig.7, Fig.8 Fig.9:

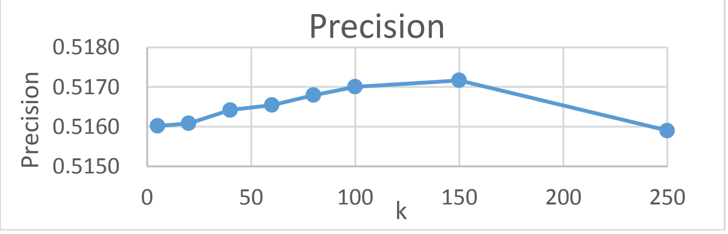

Figure 7. Precision with the first recommendation strategy.

In Fig. 7, with the increasing k, Precision continue to rise, and the highest value 0.517167 is reached in the k=150.

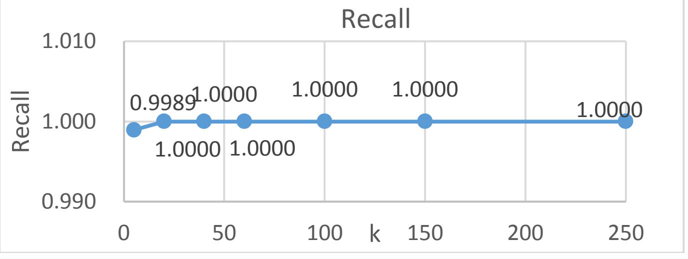

Figure 8. Recall with the first recommendation strategy.

In Fig. 8, the Recall has no obvious change, only in the k=5, Recall is 0.9989. In the other k, Recall is 1.0. To evaluate the quality of the second recommended strategy this paper retained the first recommendation strategy, and the simple collaborative filtering replaced second recommended strategy.

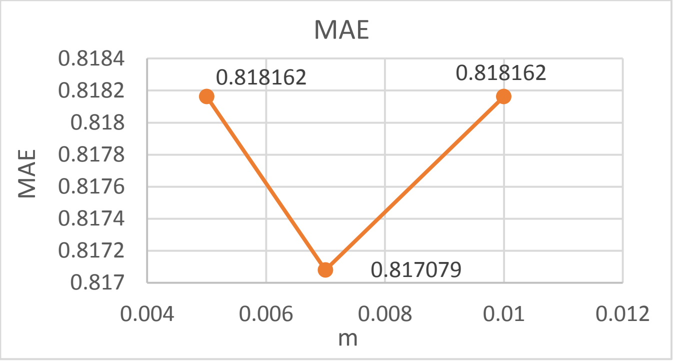

Figure 9. MAE with the second recommendation strategy.

The parameter m is set to {0.01, 0.005, 0.007}, and the experiment is processed in the same training set and test set. The MAE, Precision, Recall experimental results are as shown in Fig.9, Fig.10, Fig.11:

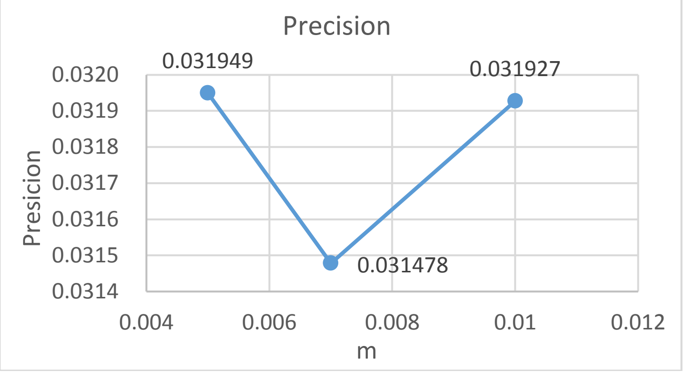

Figure 10. Precision with the second recommendation strategy.

Fig 9, Fig.10, Fig.11 indicate that the recommendation quality with the pure second recommended approach is awful, the highest Precision is only 0.031949, the largest recall is only 0.022173, the lowest value of MAE is only 0.817079.

When two recommendation strategies are not used, only using collaborative filtering algorithm, the experimental result in the same data set is MAE=0.817595, recall=0.029644, precision=0.027475.

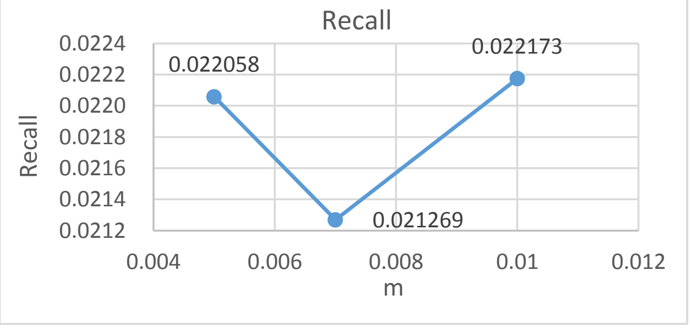

Figure 11. Recall with the second recommendation strategy.

5 Conclusions and future work

According to the validation of the two strategies mentioned above, it is proved that the two strategies are effective. A large number of experimental results shows that when using two kinds of recommendation strategies, m=100, k=0.005, MAE, Precision and Recall have reached the optimal value. And Precision, MAE, Recall compared by using any one of two combination strategies or not using any combination strategy at all have been significantly improved. Here comes to a conclusion which is that the utilization of the ICR system for IPTV recommendation system provides a very efficient solution to the problem of sparsity and cold-start.

The ICR system proposed in this paper is still insufficient. In the experimental process, due to the limitations of the SVD technology—the calculation is large, the running time is long so that the efficiency of the system is significantly affected. It can be anticipated that ICR system is facing a more serious problem of effectiveness with larger data. In this case, the author will put forward a distributed computing solution through using the Hadoop distributed computing framework to build a distributed architecture ICR system to improve the efficiency of recommendation. At the same time, it is also noted that another problem in the CF-SVD-CF algorithm, the original matrix for the first time to use collaborative filtering to predict. Although at this moment the data is denser than the data not processed by the data filter, the user-program matrix is still very sparse in a large number of programs. Therefore, the use of collaborative filtering for the original matrix prediction will make the prediction error is relatively large. The basis of this problem, the author will further use the combination method of feature value increment and the user profile. And filling a rating which is transformed into by the similarity between the program and the user profile. Then using collaborative filtering method to alleviate the sparsity of the rating matrix, and enhance the quality of recommendation ICR.$itemSet_{preferred}P_{ave}$.

References

1. Z. Yu and X. Zhou, IEEE transaction on Consumer Electronics, **50**(1), 393-399 (2004)
2. T. Isobe, M. Fujiwara, and H. Kaneta. et al., IEEE Trans Consumer Electronics, **51**(2), 665-674 (2005)
3. S. Song, H. Moustafa, and H. Afifi, IEEE Transactions on Multimedia, **14**(6), 1528-1537 (2012)
4. R. Bambini, P. Cremonesi, and R. Turrin, *Recommender Systems Handbook*, 299-331 (2011)
5. P.G. Campos, F. Díez, and I. Cantador, User Modeling and User-Adapted Interaction, **24**(1-2), 67-119 (2014)
6. N. Pervin, F. Fang, and A. Datta, et al., Acm Transactions on Management Information Systems, **3** (4), 19 (2013)
7. J.S. Breese, D. Heckerman, and C. Kadie, *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, **7**(7), 43-52 (1998)
8. B. Sarwar, G. Karypis, and J. Konstan, et al., *Proceedings of the 10th international conference on World Wide Web. ACM*, 285-295 (2001)
9. B. Arwar, G. Karypis, and J. Konstan, et al., *Processing of 2nd ACM Conference on Electronic Commerce*, 158-167 (2000)
10. A. B. Barragáns-Martínez, E. Costa-Montenegro, and J.C. Burguillo, et al., Information Sciences, **180** (22), 4290-4311 (2010)
11. G. Wang and H. Liu, Computer Engineering and Applications, **48**(7), 66-76, DOI:10.3778/ j.issn.1002-8331.2012.07.018 (2012)
12. B. Sarwar, G. Karypis, and J. Konstan, et al., In Acm Webkdd Workshop (2000)
13. H. Zhang and S. Zheng, *Consumer Electronics (ISCE 2005) Proceedings of the Ninth International Symposium on. IEEE*, 242-246 (2005)
14. S. Lee, D. Lee, and S. Lee, IEEE Transactions on Consumer Electronics, **56**(2), 1034-1042 (2010)

15. Y. Guo, Recommender system collaborative filtering algorithm and application [D], Dalian University of Technology (2008)
16. A.L. Deng, Y.Y. Zhu, B.L. Shi, Journal of Software, **14**(9), 1621-1628 (2003)
17. X. Sun. Study on Sparseness and Cold Start of Collaborative Filtration System [D], Zhejiang University (2005)
18. X. Liang, Recommended system practice [M], Beijing, People's Posts and Telecommunications Press, 2, 51, 60 (2012)