

# A Neighborhood-based Matrix Factorization Technique for Recommendation

Meng-jiao Guo<sup>1</sup> · Jin-guang Sun<sup>1</sup> ·  
Xiang-fu Meng<sup>1</sup>

Received: 29 November 2015 / Revised: 30 November 2015 / Accepted: 5 December 2015 /  
Published online: 18 December 2015  
© Springer-Verlag Berlin Heidelberg 2015

**Abstract** The data sparsity and prediction quality are recognized as the key challenges in the existing recommender Systems. Most of the existing recommender systems depend on collaborating filtering (CF) method which mainly leverages the user-item rating matrix representing the relationship between users and items. However, the CF-based method sometimes fails to provide accurate information for predicting recommendations as there is an assumption that the relationship between attributes of items is independent and identically distributed. In real applications, there are often several kinds of coupling relationships or connections existed among users or items. In this paper, we incorporate the coupling relationship analysis to capture the under-discovered relationships between items and aim to make the ratings more reasonable. Next, we propose a neighborhood-based matrix factorization model, which considers both the explicit and implicit correlations between items, to suggest the more reasonable items to user. The experimental evaluations demonstrate that the proposed algorithms outperform the state-of-the-art algorithms in the warm- and cold-start settings.

**Keywords** Recommender systems · Coupling relationship · Matrix factorization · Cold-start · Predicting

---

✉ Meng-jiao Guo  
gmj23678111@hotmail.com

Jin-guang Sun  
sunjinguang88@163.com

Xiang-fu Meng  
marxi@126.com

<sup>1</sup> School of Electronic and Information Engineering, Liaoning Technical University, Huludao 125000, China

## 1 Introduction

Recommender system (RS) is an important way to deal with the problem of information overload since it applies information filtering approaches for providing proposals for users that are suitable to their favors and tastes [1]. However, most of the existing RS perform not very well because they suffer from problems of cold start and sparsity caused by the massive growth of new items participation with no ratings. As a result, the magnitude of user/item vector could not be properly learned due to the lack of information. However, the decision-making quality of the recommender system always depends on the rating data from the users in real applications.

To deal with problem mentioned above, researchers eliciting complementary information tend to achieve high quality recommendations and its evolution are investigated from various perspectives, such as Tags [2], location information [3]. Also, there is a wealth of literatures in this area takes accounts of social friendship [4,5] while the social friendship is not always available. Furthermore, it is commonly centralized on affiliation recommendation, rarely the item recommendation in the previous study. For this reason, our approach applies the association rules of items' attributes to establish the relationship between items. Foregone literature is abundant in studies about context-aware approaches, little has been done on the premise that the attributes are i.i.d. (identically and independent distributed), the aspect of interactions and connections among them has not attract much attentions. It should be pointed out that the traditional recommender systems usually neglect the attributes correlations (including explicit or implicit connections). In contrast, the attributes are coupled (such as intra-relationship within the attribute items and inter-relationship across the attribute items) in real applications. Based on this intuition, both coupling relationships between the attributes of items and user-item rating matrix should be taken into account to facilitate the recommendation accuracy. These attributes of items can be leveraged to alleviate the cold-start problem when the ratings are sparse.

Based on the ideas mentioned above, an illustration of a movie recommendation problem in coupling relationship can be found in Fig. 1. For example, the attributes "director", "artists", and "genre" are consisted of the attributes of a movie. Different director, actor and genre jointly form the corresponding attribute values. Through the view recordings of each user can offer the information of relevant attribute values, it can establish different movies' correction by the similarity of attribute values, which are often coupled together and serve as an extra source which can provide more information to indicate why the user gives the rating.

The main reason of the problems is that the traditional recommendation strategies such as collaborative filtering (CF) generally depend on user-item rating matrix, which is usually partially filled. Latent factor models, such as matrix factorization (MF), attempt to explain the ratings by transforming both items and users to the same latent factor space. MF models are effective at apprising structure which relates to most or all items simultaneously.

This paper is prominently improved by considering of Coupled Attribute Value Similarity. In addition, item coupling process extraordinary explanatory of the existing ratings based on the empirical evidence. Therefore, combining the relevance can be potentially utilized in RS.

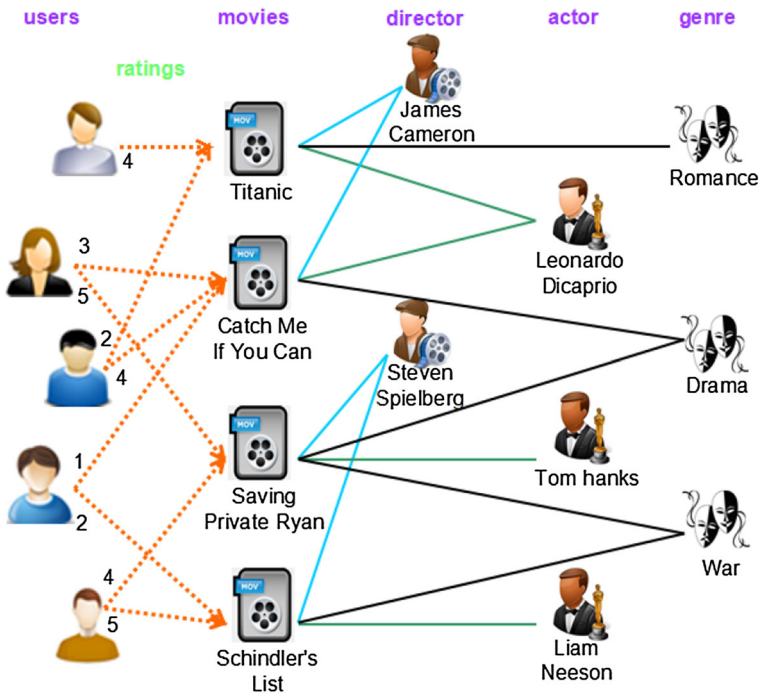


Fig. 1 A sample of the movie database

This paper enlightened by previous studies such as [5] and [6]. We present an effort to exploit the CF problem with the ratings and combination the inter relation of items. The key contributions of this paper are summarized as follows:

- We extend the item-coupling analysis method to reveal the implicit relationship between items, which enable them to effectively deal with sparse datasets;
- We focus on inferring the implicit relationship from the item coupling relationship combined with user’s subjective preferences rating scale into matrix factorization learning model;
- We conduct an extensive experimental study on two real data sets and show that the proposed methods outperform three state-of-the-art methods for item-cold start recommendation.

The structure of the remnant of the paper is organized as follows. Section 2 overviews some classical related work. Section 3 describes the item coupling relationship analysis method. Section 4 explains and analyzes the coupling Item-based MF model while Sect. 5 presents our experimental results.

## 2 Related Work

In this section, we review some classical approaches which mainly include collaborative filtering (CF) and content-based (CB) techniques in Recommender Systems.

With the emerging of CF [1, 7] in the RS field, it achieved a great accomplishment since CF methods are domain independent and only rely on historical user record without demanding the establishment of explicit profiles, as well as seizing the abstruse and difficult to profile by other means. CF method is most adept with detecting relationships between items or, alternatively, between users for generating recommendations. CF can be further categorized into the neighborhood-based and model-based methods.

In recent years, the neighborhood-based techniques have been effectively deployed and widely investigated by several researches. Neighborhood-based methods also involved user-oriented [8] and item-oriented approaches [9]. User-oriented methods mainly discern like-minded users with the similar historical actions or ratings, while item-oriented methods estimate unknown ratings on the basis of similar items that tend to be rated resemblance. User-oriented and item-oriented methods are commonly explicitly modelling the similarities of users or items or merging them together [10]. The neighborhood-based method is prevalent used since it is easier and intuitive to implement. However, although it can generate the approximately precise results, it suffers the serious limitation of scalability with the rising magnitude of users and items.

An alternative way of collaborative filtering is model-based method, which trains the observed ratings to get a well-designed model. The unknown ratings can be evaluated via the model instead of handling the original rating matrix. The bayesian hierarchical model [11, 12], clustering model [13], latent factor model [7], are the well-known examples in collaborative filtering. Among them the most widely used single model is matrix factorization (MF). The merit of MF approach is its elasticity to append some fundamental extensions to the primary model. MF techniques can be a potentially more effective method for the elusive relation data, owing to their remarkable precisely and scalability. Koren et al. did amount of relevance works as the guidance to the progress of development [7].

The mainly challenge in CF is to effectively forecast the preferences of users. However, the traditional CF mainly focuses on building the user-item matrix, meanwhile a wealth of items only rated by a small fraction of users. Therefore, leaving the majority of user-item relations unknown, the issue currently referred to the cold-start problem [14]. CF only settles the problem to a certain degree, but it cannot provide a complete solution. So, more progressive methods have been developed to fuse auxiliary information for the purpose of effectively providing the items with low popularity or new arrivals. These approaches can facilitate the personalized item recommendation performance.

CB techniques (also named “signal filtering”) [6, 15] analyze the items content to characterize its nature for depicting its profiles and establishing user-to-user or item-to-item relationships. CB method attempts to enhance the quality of recommendations exceed the capabilities of feedback-driven/purely rating RS. The most prominent issue is to discover the specific description of items’ attribute in the CB field. The character of the item data sets usually based on manual annotations by the domain experts, or even leveraging the tags from the folksonomy [16] that depict the content of the items brevity, [17] proposed that modern E-commerce platform contain review texts as items’

feature expression patterns can resolve the scarce score case. Beyond that, [4,5] put forward social network structure can also relieve the cold-start problem. In this paper, we adopt the entities' attribute feature which are more naturally and appropriate to portray.

Clearly, CF and CB method are complement with each other because of they typically deal with the same issue from different perspectives. Embedding user- and item-oriented filtering and extra information can strengthen the performance of RS framework. Several attributes similarity of user or item fusion algorithms and hybridizations has been developed by [14]. They are major in excavating the dependence of users' and item's characters and transitivity of feedback indirect neighbors in the data sets. Based on this intuition, the similarity can capture the associate of new user-item and complement the closest neighbors' predict score.

Review that numbers of modified researches has been done to ameliorate the performance of basic Matrix Factorization methods of recommender systems recently are presumed that the attributes of item are independent and identically distributed (i.i.d), and ignores the coupling relationships between items, which is not compatible with the reality situation. And there is no much work has been done to fuse relation of item attribute within model, inspired by the concept of coupled attribute value similarity (CAVS). This paper concerns on approaches which centers on item-item similarity predicts the ratings for an item on the ratings expressed by the user inclination for an item of his or her ratings on the similar items.

### 3 Coupling Relationship Analysis

Most of similarity measuring method mainly depends on the historical rating score which is usually insufficient or deals with the items of category attributes relationship separately. The general used similarity metric is Pearson correlation coefficient (PCC)

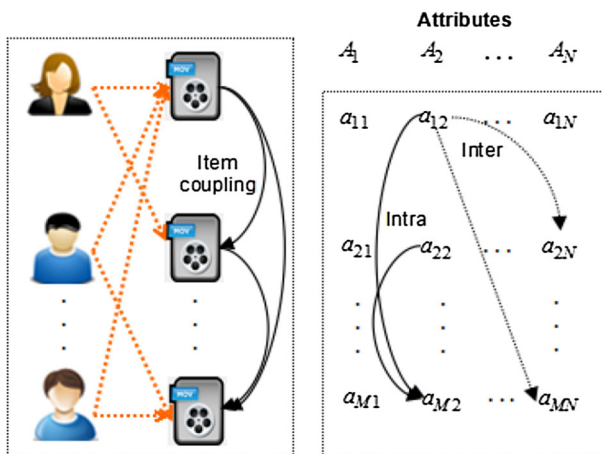


Fig. 2 Item coupling relationship

algorithm [18], which assumes there exists a linear relationship between the variables. Actually, the relationship between attributes of items should be incorporated together to measure the similarity.

In this section, to excavate the key concept of implicit relationship, we aim to leverage information of categorical attributes to unveil CAVS. CAVS is composed of both intra-coupled and inter-coupled value similarities, which can obtain the relatively accurate relationship between items. The work in [19, 20] presented a detailed analysis as showed in Fig. 2.

### 3.1 Intra-coupled Attribute Value Similarity

The intra-coupled attribute value similarity (IaAVS) explores to combine two arbitrarily items' value concurrence frequency of an attribute to reflect the similarity. The IaAVS between attribute values  $a_{ij}$  and  $a_{i'j}$  of attribute  $A_j$  is defined as follows,

$$\delta_j^{Ia}(x, y) = \frac{|g_j(x)| \cdot |g_j(y)|}{|g_j(x)| + |g_j(y)| + |g_j(x)| \cdot |g_j(y)|} \quad (1)$$

where,  $1 \leq |g_j(x)|, |g_j(y)| \leq M$ ,  $|g_j(x)|$  and  $|g_j(y)|$  are the size of subset of corresponding attribute  $A_j$  which having attribute values  $x$  and  $y$ , respectively.

### 3.2 Inter-coupled Attribute Value Similarity

The extra attention that CAVS get by the inter-coupled attribute value similarity (IeAVS) captures the interaction between different attribute. The IeAVS between attribute values  $x$  and  $y$  of attribute  $A_j$  is defined as follows.

$$\delta_j^{Ie}(x, y) = \sum_{k=1, k \neq j}^N \gamma_k \delta_{j|k}(x, y) \quad (2)$$

where,  $\gamma_k$  is the weight of attribute  $A_k$ ,  $\gamma_k \in [0, 1]$ ,  $\sum_{k=1, k \neq j}^N \gamma_k = 1$ . Since different attributes usually have different importance, they cannot be assumed equal. In this paper, we leverage entropy attribute weight assignment method, which applies the relative objective indicators to the attributes.

Based on the exemplification, we employ Shannon Entropy to depict the attribute weight. Since it is tough to gain credible subjective weights, the adoption of objective weights is demanded. Among objective weighting estimation that immensely has been used in Multi-criteria decision making (MCDM) domain is Shannon's entropy concept [21]. The notion of entropy is related to the quantity of information of a message as a statistical measure. Shannon's entropy concept is a general measure of uncertainty in information formulated in terms of probability theory. Entropy weight is an argument that accounts how much diverse alternatives approach one another with respect to a certain criteria.

Hence, this method is adaptive to be employed in our model as we will cope with category attribute. The procedure of calculating entropy weight can be formulated as follows.

$$p_k = \frac{|x_k|}{M}, \quad i = 1, \dots, M, \quad k = 1, \dots, N$$

where,  $|x_k|$  is the size of attribute value equals to  $x$  with corresponding attribute  $k$ ,  $p_{ik}$  is the probability of the occurrence of the  $k$ th attribute in the attribute set,  $h_k$  is the entropy’s abbreviation of the  $k$ th attribute, computed as:

$$h_k = -h_0 \sum_{k=1}^N p_k \ln p_k, \quad i = 1, \dots, M$$

where,  $h_0$  is the entropy constant that equals to  $(\ln M)^{-1}$ , and  $p_k \ln p_k$  is defined as 0 if  $p_k = 0$ .

The  $d_k = 1 - h_k, k = 1, \dots, N$ , is the degree of diversification.

The  $\gamma_k = \frac{d_k}{\sum_{k=1}^N d_k}$ , is the degree of magnitude of attribute  $k$ .

We compute the IeAVS of each two attribute value  $\delta_{j|k}(a_{ij}, a_{i'j})$ , given by:

$$\delta_{j|k}(x, y) = \sum_{w \in \cap} \min \{ P_{k|j}(w|x), P_{k|j}(w|y) \} \tag{3}$$

where,  $\cap$  indicates the intersection set for attribute  $A_k$  with attribute  $A_j$  of items sharing attribute values  $a_{ij}$  and  $a_{i'j}$ .  $P_{k|j}(w|x)$  is the conditional probability of attribute value  $w$  of attribute  $A_k$  based on the other attribute value  $x$  of attribute  $A_j$ , which can be computed by,

$$P_{k|j}(w|x) = \frac{|g_k(w) \cap g_j(x)|}{|g_j(x)|} \tag{4}$$

Through the analysis of IaAVS and IeAVS, the CAVS between attribute values  $a_{ij}$  and  $a_{i'j}$  of attribute  $A_j$  can be computed as follows:

$$\delta_j^A(x, y) = \delta_j^{Ia}(x, y) * \delta_j^{Ie}(x, y) \tag{5}$$

### 3.3 Item Coupling

Based on the coupled item similarity (CIS) between item  $o_i$  and  $o_j$  derived by the IaAVS  $\delta_j^{Ia}(a_{ij}, a_{i'j})$  and Inter-coupled Attribute Value Similarity  $\delta_j^{Ie}(a_{ij}, a_{i'j})$ , the CIS between two items  $o_i$  and  $o_j$  can be defined as:

$$CIS(o_i, o_j) = \sum_{j=1}^N \delta_j^A(x, y) \tag{6}$$

Here, the IaAVS within an attribute and IeAVS among different attributes are mixed together to capture the CAVS of items. Employing the item coupling similarity, we can develop our coupling item-based MF model.

## 4 Neighborhood-based MF Model

In this part, we have a brief introduction of the basic MF model. And then we import the improved coupling item-based MF model. Our algorithm is based on the Neighbor-Integrated Matrix Factorization technique introduced by [17].

### 4.1 The Basic Matrix Factorization

The basic ideal of MF approach is to decompose the scarcity of user-item matrix into a joint latent factor space of a low dimensionality  $f$ , and aims at utilizing the inner product of factorized user-specific and item-specific vectors to make further predictions in that space.

Given an  $M \times N$  rating matrix.  $E = \{r_{ui}\}$  represents  $M$  users' ratings on  $N$  items, the character of user  $u$  is depicted by the vector  $p_u \in R_f$ , which is used to gauge the affinity of the corresponding latent factors, and each item  $i$  with an item-factors vector  $q_i \in R_f$ , which is used to measure the relevance of corresponding latent factors, the matrix  $Rf$  captures the most major features of the data message, where  $f \ll \min(M, N)$ . The missing entries are obtained by multiplying the user-item feature-vector pairs correspondingly, e.g.  $\hat{r}_{ui} \approx p_u^T q_i$ . A greater inner product between a user feature vector and an item feature vector represent their tendency. The regularized squared loss is frequently used error function [4, 7].

The approximation is performed by minimizing the regularized cost function on the observed rating data [21]:

$$L(p, q) = \min_{p^*, q^*} \frac{1}{2} \sum_{(u, i) \in E} I_{ui} (r_{ui} - p_u^T q_i)^2 + \frac{\lambda_p}{2} \|p_u\|_F^2 + \frac{\lambda_q}{2} \|q_i\|_F^2 \quad (7)$$

The set  $E$  contains the  $(u, i)$  pairs of which  $r_{ui}$  is actual rating value. Where  $I_{ui}$  is the indicator function that is equal to 1 if the vectors of items rated by user  $u$  and equal 0 otherwise,  $\|\cdot\|$  denotes the Frobenius norm, the two regularization parameters  $\lambda_p, \lambda_q > 0$ . Gradient based function is able to find a local minimum. The regularizing  $\frac{\lambda_p}{2} \|p_u\|_F^2 + \frac{\lambda_q}{2} \|q_i\|_F^2$  term need to penalize the magnitudes of the parameters avoids over-fitting [3].

The traditional matrix factorization technique is context unaware method since it is primarily concentrate on the known entries, especially, the rating information is usually sparse, which suffers from poor scalability. Hence this approach is insensitive to seize a subgroup of items or users relatively similar. The overall structure will result in information loss problem.

In next section we represent some suitable variant of MF technique which improve the quality of recommendation significantly.



### 4.2 The Coupling Item-based Matrix Factorization

In this subsection, we propose our approach which leverages Coupled Attribute Value Similarity between items with the classic matrix factorization model for recommendation. It should be pointed out, although there are some differences between items, normalizing the neighbors of certain item can share similar property from some aspects which reflect the propagation of item’s trait. Namely, the item latent feature vector  $q_i$  and its neighbor feature vector tend to be resembled in the corresponding space. On top of this observation, we encapsulate the whole structure and partial information which uncover the prediction model holistically.

We have a symmetry similarity value matrix, where nodes are calculated by using CAVS [19,20]. Suppose  $W_{ij}$  is the weight of each neighbor of item  $i$ , it then can be represented by the normalized similarity as follows:

$$W_{ij} = \frac{CIS(o_i, o_j)}{\sum_{j \in \tau(i)} CIS(o_i, o_j)} \tag{8}$$

We acquire the predicted score  $R_{ui}$  by assembling the feature vector derived from the item’ own and the others are derived from their analogical neighbors in a certain degree:

$$r_{ui} \approx \hat{r}_{ui} = \alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j \tag{9}$$

where, parameter  $\alpha$  balances the influences between latent feature vector  $q_i$  and its neighbors’ latent feature vector on estimated score. The parameter  $\alpha$  implies the degree of item’s dependency on themselves and their neighborhoods,  $\tau(i)$  is the *Top-K* similar items of item  $o_i$ .

In this work, to avoid estimating the prediction beyond the limit of the reasonable scope, we map the original rating  $R_{ui}$  to the interval  $[0,1]$  by utilizing the function  $f(x) = x/R_{max}$ . Here,  $R_{max}$  denotes the maximum of rating scale. We employ the logistic function  $g(x) = 1/(1 + \exp(-x))$  proposed by [11], which makes it possible to bound the range of predictions within  $[0,1]$ . We apply our method in an item-oriented manner and such method can give a good explanation about their recommendations according to CAVS. This model is trained by the following optimization,

$$L(p, q) = \min_{p^*, q^*} \frac{1}{2} \sum_{(u,i) \in E} I_{ui} (R_{ui} - g(\alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j))^2 + \frac{\lambda_p}{2} \|p_u\|_F^2 + \frac{\lambda_q}{2} \|q_i\|_F^2 \tag{10}$$

An optimal solution of the convex problem can be obtained by least square solvers. With gradient decent, each desired parameter  $p_u, q_i$  is obtained by the local mini-

imum of the objective function to find the steepest descent direction for estimate the parameters' regional derivative respectively [6]:

$$\begin{aligned} \frac{\partial L}{\partial p_u} &= \sum_{i=1}^n I_{ui} g'(\alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j) \\ &\quad \times \left( g \left( \alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j \right) - r_{ui} \right) \\ &\quad \times \left( \alpha q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} q_j \right) + \lambda_p p_u, \end{aligned} \quad (11)$$

$$\begin{aligned} \frac{\partial L}{\partial q_i} &= \alpha \sum_{u=1}^m I_{ui} g' \left( \alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j \right) p_u \\ &\quad \times \left( g \left( \alpha p_u^T q_i + (1 - \alpha) \sum_{j \in \tau(i)} W_{ij} p_u^T q_j \right) - r_{ui} \right) \\ &\quad + (1 - \alpha) \sum_{k \in N(i)} \sum_{u=1}^m I_{uk} g' \left( \alpha p_u^T q_k + (1 - \alpha) \sum_{j \in \tau(k)} W_{kj} p_u^T q_j \right) \\ &\quad \times \left( g \left( \alpha p_u^T q_k + (1 - \alpha) \sum_{j \in \tau(k)} W_{kj} p_u^T q_j \right) - r_{uk} \right) W_{ki} p_u + \lambda_q q_i, \end{aligned} \quad (12)$$

where,  $g'(x) = \exp(x)/(1 + \exp(-x))^2$  is the derivative of logistic function of  $g(x)$ .  $N(i)$  is all the items of which the  $k$  most similar neighbors of item  $o_i$ .

The corresponding latent factors are updated in the direction opposite to the gradient proportionally, so the learning process is reformulated as follows:

$$p_u^{(iter+1)} \leftarrow p_u^{(iter)} - \eta \frac{\partial L}{\partial p_u^{(k)}} \quad (13)$$

$$q_i^{(k+1)} \leftarrow q_i^{(k)} - \eta \frac{\partial L}{\partial q_i^{(k)}} \quad (14)$$

---

**Algorithm 1** The Improved Coupling Item-based Matrix Factorization algorithm pseudo-code

---

**Input:**  $E$ : the user-item rating matrix,  $O$ : the item set,  $iter$ : the number of iterations,  $\lambda_p$ : the regularization parameter for user,  $\lambda_q$ : the regularization parameter for item,  $\alpha$ : control the effects of neighbors,  $\eta$ : the learning rate

**Output:** the user latent feature matrices  $P$  and the item latent feature matrices  $Q$

```

Step 1 : //calculate the similarity of items
1: for each  $o_i \in O$  do
2:   for each  $o_j \in O$  &  $o_i \neq o_j$  do
3:     calculate  $CIS(o_i, o_j)$  according to (6);
4:   end for
5: end for
Step 2: //Iterative learning
6: Init  $P(0)$  and  $Q(0)$  with random values and  $iter \leftarrow 0$ 
7: repeat
8:   for  $u = 1$  to  $M$  and  $i = 1$  to  $N$  do
9:     if  $I_{ui} \neq 0$  then
10:      Calculate the derivative of  $L$  respect to  $p_u, q_i$ 
        using equations (11) and (12)
11:      perform gradient step on  $p_u, q_i$  with equations
        (13) and (14)
12:     end if
13:   end for
14:    $iter++$ 
14: until error on validation set decreases or  $iter < maxIter$ 
15: return  $P, Q$ 

```

---

The parameters and results of the algorithm are expounded in the next section.

## 5 Experiments

In this section, we aim to verify the accuracy of the proposed coupling item-base matrix factorization method (CISMF). We utilize a fivefold cross-validation method for training and testing. We randomly sample each data set into five folds and pick four of them served as the training set. The rest are served as the test set for each iteration.

### 5.1 Experiments Settings

#### 5.1.1 Data Set

Experiments are deployed on two public published collaborative filtering datasets, MovieLens 100k (ML-100k) and MovieLens 1M (ML-1M). These two datasets are offered by the GroupLens research group in the Department of Computer Science and Engineering at the University of Minnesota, and are broadly adopted in the current researches.

ML-100k consisting of 100,000 ratings (1–5) derived from 943 users on 1682 movies, ML-1M offer 1 million ratings voted by 6040 users on 3900 movies. Specially, in both of the datasets all the users have rated at least 20 movies, the sparsity of the

datasets are 0.9369 and 0.9553. Apart from the historical score, it also supplies extra information about movies' attributes, containing movie genre and release year, so it is extraordinary meaningful to item-oriented recommendations.

### 5.1.2 Evaluation Metrics

We use mean absolute error (*MAE*) and root mean squared error (*RMSE*) metrics to estimate the quality of our proposed algorithms. The metric *MAE* and *RMSE* are defined as follows, respectively.

$$MAE = \frac{\sum_{(u,i) \in r_{test}} |r_{ui} - \hat{r}_{ui}|}{|r_{test}|}, \quad RMSE = \sqrt{\frac{\sum_{(u,i) \in r_{test}} (r_{ui} - \hat{r}_{ui})^2}{|r_{test}|}}$$

where,  $r_{ui}$  is the real rating,  $\hat{r}_{ui}$  is the predicted rating given by user  $u$  on item  $i$  and  $r_{test}$  is the number of all user-item pairs in the test set. Smaller *MAE* or *RMSE* represents superior prediction accuracy.

## 5.2 Comparison with Other Method

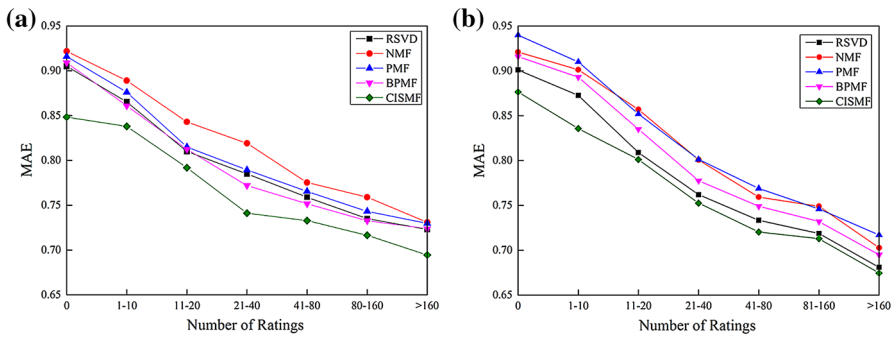
To show the prediction accuracy of our method, we consider the following three representative approaches as comparison partners.

- A. RSVD: regularization singular value decomposition is introduced in [22], which is a classic baseline model.
- B. NMF: non-negative matrix factorization is represented by [23], which restrict the latent feature non-negative update during the learning process.
- C. PMF: probabilistic matrix factorization is proposed by [11]. It is a well-known method used in traditional recommender systems.
- D. BPFM: Bayesian Probabilistic Matrix Factorization is proposed by [12], the method efficiently employs Markov chain Monte Carlo methods.

The parameter settings of our method are  $\alpha = 0.6$ ,  $Top-k = 10$ ,  $\lambda_p = \lambda_q = 0.001$ , and  $d = 5$  in the experiments. As Table 1 reported, it summarizes the results on testing data that we can see our results in the last column outperform the other Methods on two commonly used data sets. The bigger size of dimension may bring more noise into the model during learning procedure. The improvements are significant, which

**Table 1** Results of comparative study on the MovieLens datasets

Metrics	Movielens100K					Movielens1M				
	RSVD	NMF	PMF	BPMF	CISMF	RSVD	NMF	PMF	BPMF	CISMF
<i>MAE</i>	0.7433	0.7724	0.7522	0.7465	0.7279	0.6885	0.7286	0.7306	0.7023	0.6814
<i>RMSE</i>	0.9473	0.9874	0.9667	0.9533	0.9268	0.8670	0.9203	0.9234	0.8907	0.8592



**Fig. 3** Comparison on different items. **a** MovieLens100K **b** MovieLens1M

reveals promising orientation of recommendations. In the following we explore the other aspect factors in more detail and we only display the performance in *MAE*.

### 5.3 Validation on Cold Start Items

The main central issue of the recommender systems area is cold-start problem, very few approach investigates the few item ratings. In this paper, we compare our method with other methods comprehensively. The number distribution of items are divided into 7 groups: “=0”, “1–10”, “11–20”, “21–40”, “41–80”, “81–160” and “>160”, implying the items have received how many ratings.

Figure 3 demonstrates the quantitative results for items presented with respect to different categories. As shown in the figures, CISMf achieves the best results, which indicates that the survey of considering the coupling relationship is effective.

### 5.4 Validation of Parameter

The parameter  $\alpha$  controls the influence of the items themselves and their neighbors should be merged into the predicted rating. We explore the varying tendency of our proposed approach by adjusting the value  $\alpha$  from 0 to 1. Let us consider the extreme condition,  $\alpha = 1$  means we only mine the rating matrix, making it equals to RSVD [9], whereas  $\alpha = 0$  implies we predict the rating not rely on itself instead of its neighbors to get the prediction.

From Fig. 4, we can see that the performance of fusing similar neighborhoods via CISMf are provided when  $\alpha$  changes. It performed optimal value at  $\alpha = 0.5$  on MovieLens 100K and  $\alpha = 0.6$  on MovieLens 1M, respectively. It suggests that the neighbors’ feature is valuable for our model.

### 5.5 Validation of Size of Neighborhood

The size of *Top-k* determines the number of similar items as well as affects the model performance. We investigate the size in range of 10–50 and 10–100 with the interval 10, 20, we set the parameter  $\alpha = 0.5$  and  $\alpha = 0.6$  in MovieLens100K and MovieLens1M, prescriptively.

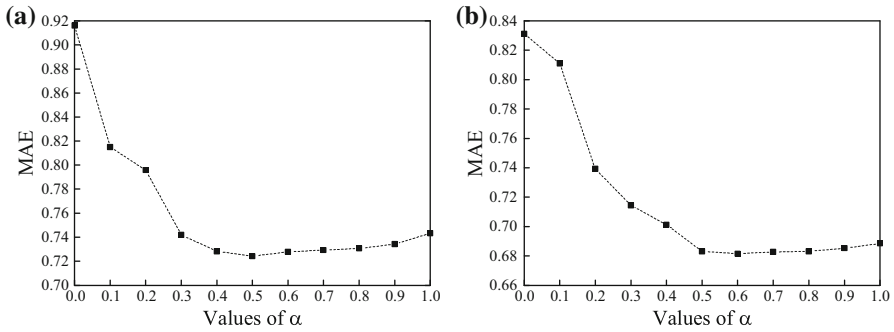


Fig. 4 Impact of parameter  $\alpha$ . a MovieLens100K b MovieLens1M

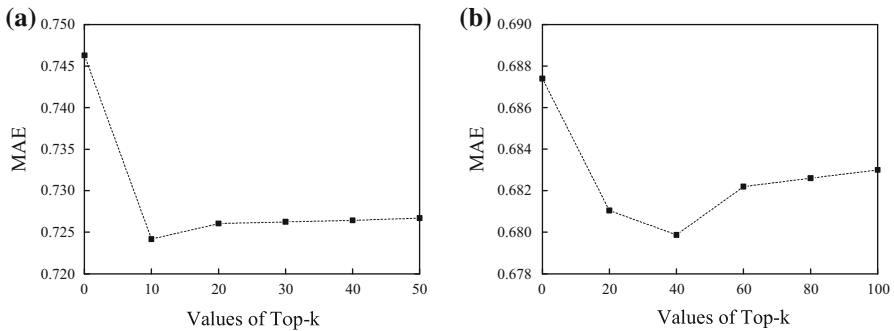


Fig. 5 Impact of size of neighborhood. a MovieLens100K b MovieLens1M

Figure 5a shows that the impact of size of neighborhood on MAE in MovieLens100K. From the figure, we can observe the deviation reach the minimum value happens for  $Top-k = 10$ , along with the value of  $Top-k$  increasing in range of 10 to 50, the MAE slightly rise. As Fig. 5b reflected, the influence of size of neighborhood on MAE in MovieLens1M. Values of  $Top-k$  lie in the range of 20–100 with step size of 20, the MAE values does not behave evident fluctuation begin with  $Top-k = 40$ . Through our analysis, it can be manifested that too few neighbors may not provide enough information while too many neighbors may bring some uncorrelated information, both of them can result the decrease of accuracy.

## 6 Conclusion

In this paper, we addressed the issues of cold start problem for new and receive few ratings’ items which is not well studied. In terms of the intuition that items’ attribute information can boost the accuracy of prediction, we have employed a novel coupling similarity measure fusing into matrix factorization for recommender system. According to our analysis and experiments, we capture coupling relationship serves as better information providers for similar items.

In the future research, we will collect more dataset with correlative attributes and use it to enhance our algorithm. Meanwhile, the cold-start user we haven't consider in this paper, it is intriguing us to consider rich social relationship in the recommendation framework, and besides, we plan to further investigate the new algorithm.

**Acknowledgments** Thanks to the support by Natural Science for Youth Foundation of China (No. 61003162) and the Young Scholars Growth Plan of Liaoning (No. LJQ2013038).

## References

1. Linden G, Smith B, York J (2003) Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Comput* 7(1):76–80
2. Jaschke R, Marinho L, Hotho A, Schmidt L, Stumme G (2007) Tag recommendations in folksonomies. *Proceedings of the 11th conference on european conference on principles and practice of knowledge discovery in databases*, pp 506–514
3. Levandoski J, Sarwat M, Eldawy A, Mokbel F (2012) LARS: a location-aware recommender system. *Proceedings of the 28th conference on on data engineering*, pp 450–461
4. McAuley J, Leskovec J (2013) From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. *Proceedings of the 22th international conference on world wide web*, pp 897–908
5. Ma H, King I, Lyu MR (2009) Learning to recommend with social trust ensemble. *Proceedings of the 32th conference on research and development in information retrieval*, pp 203–210
6. Ma H, Zhou D, Liu C (2011) Recommender system with social regularization. *Proceedings of the 4th conference on web search and data mining*, pp 287–296
7. Koren Y (2010) Collaborative filtering with temporal dynamics. *Commun ACM* 53(4):89–97
8. Middleton SE, Shadbolt NR, De DC (2004) Rouse: ontological user profiling in recommender systems. *ACM Trans Inf Syst* 22(1):54–88
9. Paterek A (2007) Improving regularized singular value decomposition for collaborative filtering. *Proceedings of the 13th international conference on knowledge discovery and data mining*, pp 5–8
10. Salakhutdinov R, Mnih A (2008) Bayesian probabilistic matrix factorization using markov chain monte carlo. *Proceedings of the 25th conference on international conference on machine learning*, pp 880–887
11. Sarwar M, Karypis G, Konstan J, Riedl J (2002) Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. *Proceedings of the 5th international conference on computer and information technology*, p 1
12. Salakhutdinov R, Mnih A (2007) Probabilistic matrix factorization. *Proceedings of the 20th conference on neural information processing systems foundation*, pp 1257–1264
13. Sarwar B, Karypis G, Riedl J (2001) Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international conference on world wide web*, pp 285–295
14. Gantner Z, Drumond L, Freudenthaler C, Rendle S, Schmidt-Thieme L (2010) Learning attribute-to-feature mappings for cold-start recommendations. *Proceedings of the 10th international conference on data mining*, pp 176–185
15. Balabanovic M, Shoham Y (1997) Fab: content-based collaborative filtering. *Commun ACM* 40(3):66–72
16. Hotho A, Jaschke R, Schmitz C, Stumme G (2006) FolkRank: a ranking algorithm for folksonomies. *LWA* 1:111–114
17. Li FF, Xu GD, Cao LB (2014) Coupled item-based matrix factorization. *Proceedings of the 15th international conference on web information systems engineering*, pp 1–14
18. Breese JS, Heckerman D, Kadie C (1998) Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the 14th conference on uncertainty in artificial intelligence*, pp 43–52
19. Cao L, Ou Y, Yu PS (2012) Coupled behavior analysis with applications. *IEEE Trans Knowl Data Eng* 24(8):1378–1392
20. Wang J, De Vries AP, Reinders MJT (2006) Unifying user-based and item-based collaborative filtering approaches by similarity fusion. *Proceedings of the 29th conference on research and development in information retrieval*, pp 501–508

21. Lotfi H, Fallahnejad R (2010) Imprecise shannon's entropy and multi attribute decision making. *Entropy* 12(1):53–62
22. Nguyen JJ, Zhu M (2013) Content-boosted matrix factorization techniques for recommender systems. *Stat Anal Data Mining* 6(4):286–301
23. Lee DD, Seung HS (2001) Algorithms for non-negative matrix factorization. *Proceedings of the 14th conference on advances in neural information processing systems*, pp 556–562