Xin X, Lin CY, Wei XC *et al.* When factorization meets heterogeneous latent topics: An interpretable cross-site recommendation framework. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 30(4): 917–932 July 2015. DOI 10.1007/s11390-015-1570-x

When Factorization Meets Heterogeneous Latent Topics: An Interpretable Cross-Site Recommendation Framework

Xin Xin [1,*] (辛 欣), *Member, CCF, ACM, IEEE*, Chin-Yew Lin [2] (林钦佑), *Member, ACM, IEEE*
Xiao-Chi Wei [1] (魏骁驰), *Member, CCF, ACM*, and He-Yan Huang [1] (黄河燕), *Member, ACM*

[1] *Beijing Engineering Research Center of High Volume Language Information Processing & Cloud Computing, School of Computer Science, Beijing Institute of Technology, Beijing 100081, China*
[2] *Microsoft Research Asia, Beijing 100080, China*

E-mail: xxin@bit.edu.cn; cyl@microsoft.com; {wxchi, hhy63}@bit.edu.cn

Received November 15, 2014; revised March 20, 2015.

Abstract Data sparsity is a well-known challenge in recommender systems. Previous studies alleviate this problem by incorporating the information within the corresponding social media site. In this paper, we solve this challenge by exploring cross-site information. Specifically, we examine: 1) how to effectively and efficiently utilize cross-site ratings and content features to improve recommendation performance and 2) how to make the recommendation interpretable by utilizing content features. We propose a joint model of matrix factorization and latent topic analysis. Heterogeneous content features are modeled by multiple kinds of latent topics. In addition, the combination of matrix factorization and latent topics makes the recommendation result interpretable. Therefore, the above two issues are simultaneously solved. Through a real-world dataset, where user behaviors in three social media sites are collected, we demonstrate that the proposed model is effective in improving recommendation performance and interpreting the rationale of ratings.

Keywords collaborative filtering, recommender system, topic analysis

1 Introduction

With the explosive growth of web information, recommender systems have achieved great success in solving the information overload problem. It helps to find information items that are likely to attract users, such as movies, music, or products. Collaborative filtering (CF) serves as the main recommendation technique. It predicts the interest of each individual user, by analyzing the preference pattern of similar users. Particularly, the factorization-based approach is a kind of competitive model, widely utilized in both competitions[1] and research communities[2-3].

Data sparsity is a typical challenge for recommender systems. It means that the density of the user-item rating matrix is extremely low in many cases (e.g., 1.18% in the well-known Netflix dataset[4]). Thus when a user/item has very few ratings, it is difficult to have accurate preference predictions. For new users/items, of which no ratings are observed, factorization-based models will not work. This is also known as the cold start problem. Previously, researchers incorporated more ratings[5] and content features[6-7] in the corresponding site, to alleviate this problem.

In this paper, we explore information from multiple social media sites rather than a single one, in order to tackle the data sparsity challenge. A typical scenario is shown in Fig.1. The active task is to predict the ratings of Foursquare check-ins by each user. For the user in Fig.1, though he/she rarely has ratings

Regular Paper
Special Section on Social Media Processing
This work was supported by the National Basic Research 973 Program of China under Grant No. 2013CB329605, the National Natural Science Foundation of China under Grant Nos. 61300076 and 61375045, the Ph.D. Programs Foundation of Ministry of Education of China under Grant No. 20131101120035, and the Excellent Young Scholars Research Fund of Beijing Institute of Technology.
*Corresponding Author

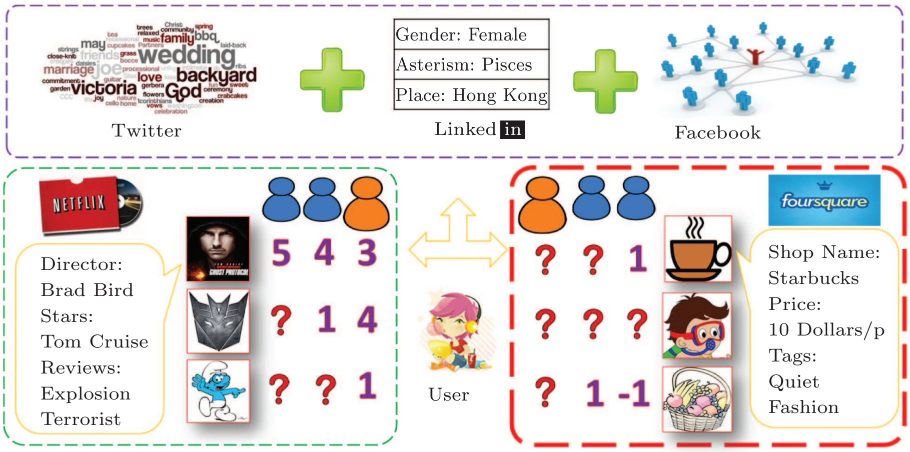


Fig.1. Typical cross-site recommendation scenario.

in Foursquare, he/she has many ratings on movies in Netflix. Thus one question is whether these cross-site ratings can be leveraged to improve the active rating predications. Besides, many kinds of content features can be collected. For example, his/her demographic information can be obtained through Linkedin; his/her focused topics can be summarized by the word cloud from tweets on Twitter; and his/her social relationships can be found on Facebook. For check-ins, we can collect their genres, average prices, and tags. For movies, we can collect their directors, players, and reviews. Therefore, another issue is whether these content features are effective in improving active recommendation. Unfortunately, explorations of the cross-site information have rarely been thoroughly studied previously.

When considering the methodology to integrate cross-site information, one limitation of most previous factorization-based CF approaches is that rating predictions (presented by the inner product of user and item latent vectors) are uninterpretable. This interpretation issue refers to inferring the rationale of generating a rating. For example, a man and a woman both give high ratings to the movie "Mission Impossible". The man likes the movie because of the exciting actions, but the woman is attracted by the handsomeness of actor "Tom Cruise". Understanding such rationale behind a rating is an important issue[8]. It will make the recommendation result more convincing (avoiding the case of some algorithm occasionally only satisfying some evaluation measures) and better, especially for cross-site recommendations, where there are a number of content features in nature. In the above case, by exploring the rationale, more action-attractive movies will be delivered to the man, and more actor-attractive movies will be delivered to the woman. In addition, when a movie investigator makes decisions, the statistics of such information can be utilized to balance whether to invite a good-looking actor or a professional action director, in order to obtain the best sale. Recently, this interpretation issue has drawn lots of attention[9-11]. Most of these studies only incorporate a single kind of latent topics, corresponding to only a single kind of feature to be incorporated naturally. When there are multiple features, presenting all of them using only one topic model might limit recommendation performance and interpretation ability.

In order to solve the above uninterpretable limitation of traditional CF models when incorporating content features, we propose a recommendation framework by considering the following motivations.

1) The rationale of a rating can be interpreted by the matching of content features from both the user side and the item side. For example, a user tagged with "rebellious" might prefer a movie tagged with "hero", because the user tag "rebellious" can match the movie tag "hero".

2) A content feature has unique matching pattern with other features of users/items. For example, the user tag "rebellious" matches the movie tag "hero" stronger than the movie tag "romantic"; but the user tag "fashion" matches "romantic" stronger than "hero".

3) The generation of a rating comes from the combination of matching strengths among content features. For example, a movie rating by a user is determined by simultaneously considering matchings among user features (e.g., tags, gender) and movie features (e.g., genres, actors).

By considering the above motivations, in the proposed model, each user/item is presented by a latent vector, which is a combination of its topic proportions of multiple content features. The topic proportion of each kind of feature is described by its unique probabilistic latent semantic indexing (PLSI)[12] model. The topics from heterogeneous spaces are assumed to be mapped into the same space by linear transformations. The proposed framework jointly models matrix factorization and topic analysis, which simultaneously solves the recommendation task and the interpretation task. Experimental verifications in a real-world cross-site dataset demonstrate that the proposed framework is effective in both the rating prediction task and the interpretation task. Meanwhile, it is also shown that utilizing the cross-site information does alleviate the data sparsity problem effectively. Through complexity analysis, the calculation cost of the proposed approach scales linearly with the number of observations. Thus it can be applied to large-scale data applications.

2 Related Work

The novelty of this work lies in two aspects. First, we explore cross-site information in solving the data sparsity problem, whereas most previous studies only utilize in-site information. The word "in-site" means all the information is collected in the same corresponding site. Second, we propose utilizing multiple PLSI models to present multiple kinds of content features. Previous studies, however, only utilize a single topic model. Thus in this section, we review several recommendation techniques, including 1) traditional collaborative filtering algorithms, 2) collaborative filtering with in-site features incorporated, and 3) recommendation interpretation algorithms.

2.1 Traditional CF Techniques

Traditionally, CF models are divided into memory-based (also called neighborhood-based) ones and model-based ones. When predicting the rating of a user-item pair, memory-based algorithms[13-17] find similar users/items for the active user/item as their neighborhood, by directly calculating corresponding similarities; model-based algorithms, on the other hand, predict ratings by training a predefined model. Typical models include aspect model[18], flexible mixture model[19], hierarchical model[20], non-parametric Bayesian model[21], restricted Boltzmann machines[22], etc. Recently, factorization-based recommendation models[3,23-25] have attracted the most attention, due to their effectiveness and efficiency, especially when applied to large-scale dataset. Probabilistic matrix factorization (PMF)[3] is one of the most competitive models among factorization-based methods, proposed by Salakhutdinov and Mnih. Followed by this method, Gaussian-Wishart priors are utilized to make the model in a full Baysian manner[23], which obtains better results. Koren *et al.* illustrated several promising improvements on the matrix factorization by integrating implicit feedback, temporal patterns, and confidence estimation[2]. Although the above methods have achieved great success in improving recommendation performance, most of them utilize ratings only, with data sparsity remaining a challenge for new users/items.

2.2 Features: In-Site vs Cross-Site

To alleviate data sparsity, researchers incorporate a range of content features into the CF algorithms. Such features include user demographics, item genres[6], tags[26], social relationships[27], etc. Recently, due to its advantage in effectiveness and efficiency, integrating features into factorization-based recommendation algorithms has drawn a lot of attention. Techniques are divided into 1) sharing latent vectors of feature-alike users/items[1,28-31], 2) learning latent vector priors through feature regression[6,32], 3) regularizing users/items with similar features[33], etc. Through empirical study[1,28], when the feature space is large, reducing its dimensionality before integration can achieve better performance, compared with directly incorporating original feature values. As discussed before, one limitation of these factorization-based models is that latent vectors have no presentive meanings. Thus the recommendation is uninterpretable. In this paper, we solve this limitation by jointly modeling matrix factorization and heterogeneous topic analysis. Besides content features, more ratings are also incorporated into CF algorithms[5,34-35]. Hu *et al.*[5] proposed to utilize a user's book ratings to enhance his/her music rating predictions. Although books and music belong to different domains, they are still in the same site. In this paper, we investigate the effectiveness of cross-site ratings in improving active rating predictions. The ratings of a user on multiple sites are explored.

2.3 Interpretation: Homogenous Topics vs Heterogeneous Topics

Recently, interpreting the latent vectors of factorization-based collaborative filtering has been a popular research issue[9-11,36]. One identical aspect among all methods is the combination of matrix factorization and topic analysis. Topic models[12,37-38] are utilized as interpretations. Agarwal and Chen proposed an fLDA model[9]. The latent vector of items is replaced by the latent topic of the item's content features. Wang and Blei[11] extended this model by adding a latent bias vector into the item latent vectors. In these two methods, the topic analysis is processed in a supervised manner jointly with the optimization of matrix factorization. McAuley and Leskovec[10] proposed discovering topics that are correlated with the hidden factors directly by tightening up the latent vectors of matrix factorization and latent topics of topic analysis. The limitation of these methods is that only a single kind of latent topic is considered. If only incorporating a single kind of feature, these models can work well. When incorporating multiple features, it is concerning that presenting all the features with only one topic model might limit performance and interpretation ability of the recommendation model. Thus we extend the model of Wang and Blei by utilizing multiple topic models.

3 Problem Definition

In this section, we define the cross-site recommendation and the recommendation interpretation problems. Suppose a user's ratings from different social media sites are collected. There are then two rating matrices, which share the same set of users. Tables 1 and 2 illustrate an example. Table 1 contains ratings from a movie review site; and Table 2 contains ratings from a point-of-interest (POI) review site. Besides ratings, heterogeneous content features are also collected, such as users' demographic information, movies' content data, and POIs' content data. The word "heterogeneous" means the value spaces of features (or latent topics in Section 4) are different. In this paper, we mainly deal with bag-of-words content features, as this kind of features has been shown to cover a great range of all features in practice[1]. The bag-of-words features are further divided into two classes, including category features and word-like features. As shown in Table 3, in category features, an entity belongs to at most one kind of feature value, such as user gender, user age, or taxonomy; but in word-like features (in Table 4), an entity can have different weights on multiple feature values, like user tags and movie actors. In many cases, social information can also been utilized as word-like features. For example, in the micro-blog's follower-followed relationship, the followed can be seen as "words", and a user's all followed can be seen as a "document". Such treatment of social information is also effective in improving the recommendation result[1].

Table 1. Cross-Site Rating Matrix — User-Movie Matrix

User	Movie 1	Movie 2	Movie 3
1	5		3
2		3	1

Table 2. Cross-Site Rating Matrix — User-POI Matrix

User	Movie 1	Movie 2	Movie 3
1		4	1
2	2	3	

Table 3. Bag-of-Words Features — Category Feature

User	Male	Female
1	1	
2		1

Table 4. Bag-of-Words Features — Word-Like Feature

User	Tag 1	Tag 2	Tag 3	Tag 4
1	3		9	
2		1		

By defining the cross-site rating matrices and content features, the problem of cross-site recommendation targeted in this paper is to investigate how to effectively and efficiently predict the missing values of the active user-item rating matrix, by employing cross-site ratings and content features.

The problem with recommendation interpretation is defined by three subtasks in this paper as follows.

1) Between two arbitrary kinds of content features h_a and h_b, let W_a denote value space of feature h_a, and let W_b denote value space of feature h_b. For arbitrary feature value w_a in W_a, which feature values in W_b can w_a match? For example, for users with the tag "rebellious", what kind of movie actors do they prefer?

2) Between a user/item i and an arbitrary kind of content feature h_a, which feature values in W_a can user/item i match? For example, for an individual

user, regardless of his/her content feature, what kind of movie player does he/she prefer?

3) In determining a rating, which features have the most impact? For example, when a user gives a high rating to a movie, is it because he/she likes those sorts of actors or those sorts of directors?

4 Matrix Factorization Through Heterogeneous Latent Topics

4.1 Generative Process

The proposed model, Matrix Factorization Through Heterogeneous Latent Topics (MFHLT), is a generative model as shown in Fig.2. The corresponding notations are shown in Table 5. Let $\boldsymbol{R}^{(uv)}$ be an $N^{(u)} \times N^{(v)}$ rating matrix, whose element $r_{ij}^{(uv)}$ is the rating of item j on site v by user i. We have converted the original value by $r_{ij}^{(uv)} = r_{ij}^{(uv)} - b_{ij}$ to remove the user/item's personal bias, where b_{ij} is equivalent to $b_0 + b_i + b_j$ as defined in [2]. This matrix is factorized by two matrices $\boldsymbol{Q}^{(u)}$ and $\boldsymbol{Q}^{(v)}$, where $\boldsymbol{Q}^{(u)}$ is an $N^{(u)} \times m$ matrix with each row $\boldsymbol{q}_i^{(u)}$ denoting an m-dimensional latent feature vector of user i, and $\boldsymbol{Q}^{(v)}$ is an $N^{(v)} \times m$ matrix with each row $\boldsymbol{q}_j^{(v)}$ denoting an m-dimensional latent feature vector of item j in site v. Each element $r_{ij}^{(uv)}$ follows a Gaussian distribution with the mean $(\boldsymbol{q}_i^{(u)})^{\mathrm{T}}\boldsymbol{q}_j^{(v)}$ and the variance $\frac{1}{\gamma^{(uv)}}$. In a similar manner, $\boldsymbol{R}^{(us)}$ is an $N^{(u)} \times N^{(s)}$ matrix, whose element $r_{ik}^{(us)}$ is the rating of item k on site s by user i. Each element $r_{ik}^{(us)}$ follows a Gaussian distribution with the mean $(\boldsymbol{q}_i^{(u)})^{\mathrm{T}}\boldsymbol{q}_k^{(s)}$ and the variance $\frac{1}{\gamma^{(us)}}$.

Let $\rho \in \{u, v, s\}$ for the convenience of presentation. In the MFHLT model, each latent feature vector $\boldsymbol{q}^{(\rho)}$ is a combination of a bias vector and the topic proportions of multiple features.

The bias vector, denoted by $\boldsymbol{\varepsilon}^{(\rho)}$, is an m-dimensional vector, referring to the personal bias for an individual user/item.

The multiple features are modeled by PLSI for the purposes of interpretation and feature dimensionality reduction, following the ideas of previous work[11]. But unlike this previous work[11], which only deals with a single feature (words in scientific articles), in our case, we face multiple heterogeneous features, e.g., movie tags, movie players. Thus we model each kind of feature by a unique PLSI model, which has its own feature value space and latent topic space. With the PLSI model, each kind of word-like feature for a user/item can be converted into a topic proportion vector. For category features, it is converted into a proportion vector without using PLSI, by setting a value of 1 in the corresponding dimension, and setting a value of 0 in other dimensions. For convenience, we treat them the same as word-like features in defining notations.

Table 5. Notations

Symbol	Description
$N^{(\rho)}$	Number of users/items
$M^{(\rho)}$	Total feature number of users/items
h	Precisely $h^{(\rho)}$, feature index for each user/item
$T_h^{(\rho)}$	Topic number of the h-th feature
l	Precisely $l_{ih}^{(\rho)}$, word index of the h-th feature for the i-th user/item
$\boldsymbol{\theta}_{ih}^{(\rho)}$	Topic proportion of the h-th feature for the i-th user/item
$z_{ihl}^{(\rho)}$	Sampled topic of the l-th "word" in the h-th feature for the i-th user/item
$w_{ihl}^{(\rho)}$	The l-th "word" in the h-th feature for the i-th user/item
$\varphi_{ht}^{(\rho)}$	Word proportion of the t-th topic for the h-th feature of users/items
$\boldsymbol{q}^{(\rho)}$	Latent feature vector of users/items
$\boldsymbol{\varepsilon}^{(\rho)}$	Preference bias vector of users/items
$\frac{1}{\lambda^{(\rho)}}$	Variance of latent preference bias vector
r	rating-b_{ij}
$\frac{1}{\gamma}$	Variance of ratings
$\boldsymbol{A}_h^{(\rho)}$	Transformation matrix of the h-th feature
$f_{ih}^{(\rho)}$	Weight for the h-th feature of user/item i

The combination of topic proportions of heterogeneous features is a challenging issue. First, the topic spaces of these features are different. Second, their topic numbers are different. Third, for a user/item, topic proportion vectors are different from latent vectors in the matrix factorization, as the former ones have to be normalized positive vectors. By considering the above issues, in this paper, we assume that topic proportions from heterogeneous feature spaces can be converted into the same feature space as the matrix factorization by linear transformation. We utilize a transformation matrix $\boldsymbol{A}$ to denote the process. For example, the topic proportion of a user feature $\boldsymbol{\theta}_{ih}^{(u)}$ is transformed by multiplying $\boldsymbol{A}_h^{(u)}$ as $\boldsymbol{A}_h^{(u)}\boldsymbol{\theta}_{ih}^{(u)}$, where $\boldsymbol{A}_h^{(u)}$ is an $m \times T_h^{(u)}$ matrix. After the transformation, the transformed topic proportions of multiple features for a user/item are linearly combined with the bias vector by multiplying a personal weight bias $f_{ih}^{(\rho)}$.

Based on the above discussions, the generative process of the proposed MFHLT model is as follows, where $\mathcal{D}$ denotes the Dirichlet distribution, $\mathcal{N}$ denotes the

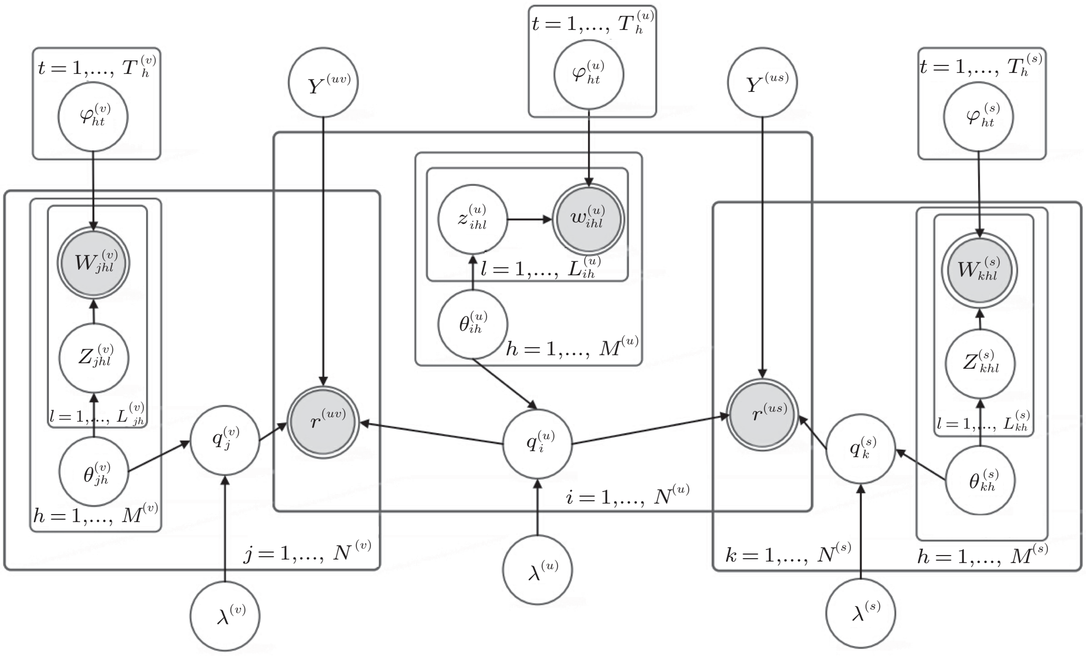


Fig.2. Graphical model for matrix factorization through heterogeneous latent topics. For the convenience of presentation, the topics of different features of a user/item have been aggregated in one plate, though they do not follow the i.i.d. property. The hyper-parameters of LDA are also omitted.

Gaussian distribution, and $\mathcal{M}$ denotes the Multinomial distribution.

1) For each kind of user/item feature $h \in \{1, ..., M^{(\rho)}\}$,

- For each of its topic t, draw $\varphi_{ht}^{(\rho)} \sim \mathcal{D}(\boldsymbol{\beta}_h^{(\rho)})$.
- Draw a transformation matrix $\boldsymbol{A}_h^{(\rho)} \sim \mathcal{N}(0, \frac{1}{\lambda^{(A)}}\boldsymbol{I})$, where $\boldsymbol{I}$ is an identity matrix.

2) For each user/item i,

- Draw a latent bias vector $\boldsymbol{\varepsilon}_i^{(\rho)} \sim \mathcal{N}(0, \frac{1}{\lambda^{(\rho)}}\boldsymbol{I})$.
- For each kind of user/item feature h,
  - Draw its topic proportions $\boldsymbol{\theta}_{ih}^{(\rho)} \sim \mathcal{D}(\boldsymbol{\alpha}_h^{(\rho)})$, where $\boldsymbol{\alpha}$ is a vector of hyperparameters.
  - For each word l,
    * Draw its topic $z_{ihl}^{(\rho)} \sim \mathcal{M}(\boldsymbol{\theta}_{ih}^{(\rho)})$.
    * Draw the word $w_{ihl}^{(\rho)} \sim \mathcal{M}(\boldsymbol{\varphi}_{hz_{ihl}^{(\rho)}}^{(\rho)})$.
- Draw a weight vector $\boldsymbol{f}_i^{(\rho)} \sim \mathcal{N}(0, \frac{1}{\lambda^{(f)}}\boldsymbol{I})$.
- Set the user/item latent factor $\boldsymbol{q}_i^{(\rho)} = \boldsymbol{\varepsilon}_i^{(\rho)} + \sum_{h=1}^{O} f_{ih}^{(\rho)} \boldsymbol{A}_h^{(\rho)} \boldsymbol{\theta}_{ih}^{(\rho)}$.

3) For each user-item pair (i, j) in site v, draw the rating $r_{ij}^{(uv)} \sim \mathcal{N}((\boldsymbol{q}_i^{(u)})^{\mathrm{T}} \boldsymbol{q}_j^{(v)}, \frac{1}{\lambda^{(uv)}})$.

4) For each user-item pair (i, k) in site s, draw the rating $r_{ik}^{(us)} \sim \mathcal{N}((\boldsymbol{q}_i^{(u)})^{\mathrm{T}} \boldsymbol{q}_k^{(s)}, \frac{1}{\lambda^{(us)}})$.

4.2 Explanations

If we expand the calculation of $(\boldsymbol{q}_i^{(u)})^{\mathrm{T}} \boldsymbol{q}_j^{(v)}$, it is composed of four terms as

$$
\begin{aligned}
&(\boldsymbol{q}_i^{(u)})^{\mathrm{T}} \boldsymbol{q}_j^{(v)} \\
&= \left(\sum_{h=1}^{M^{(u)}} f_{ih}^{(u)} \boldsymbol{A}_h^{(u)} \boldsymbol{\theta}_{ih}^{(u)}\right)^{\mathrm{T}} \left(\sum_{h=1}^{M^{(v)}} f_{jh}^{(v)} \boldsymbol{A}_h^{(v)} \boldsymbol{\theta}_{jh}^{(v)}\right) + \\
&(\boldsymbol{\varepsilon}_i^{(u)})^{\mathrm{T}} \left(\sum_{h=1}^{M^{(v)}} f_{jh}^{(v)} \boldsymbol{A}_h^{(v)} \boldsymbol{\theta}_{jh}^{(v)}\right) + \\
&(\boldsymbol{\varepsilon}_j^{(v)})^{\mathrm{T}} \left(\sum_{h=1}^{M^{(u)}} f_{ih}^{(u)} \boldsymbol{A}_h^{(u)} \boldsymbol{\theta}_{ih}^{(u)}\right) + \\
&(\boldsymbol{\varepsilon}_i^{(u)})^{\mathrm{T}} \boldsymbol{\varepsilon}_j^{(v)}.
\end{aligned}
$$

The calculation of $(\boldsymbol{q}_i^{(u)})^{\mathrm{T}} \boldsymbol{q}_k^{(s)}$ can be expanded similarly.

The last term is the personal bias values for unique user-item pairs. It has the same meaning as traditional models[3]. By analyzing the first three terms, the insight explanations of how to generate a prediction can be demonstrated.

The first term can be explained as how well topics of different features match, which corresponds to the first subtask in the recommendation interpretation problem. If two features match well, then a positive rating bias value will be generated; otherwise, a negative rating bias will be generated. For example, let h_a be the user tag feature, and let h_b be the movie player feature. Its corresponding term $(f_{ih_a}\boldsymbol{A}^{(u)}_{h_a}\boldsymbol{\theta}^{(u)}_{ih_a})^{\mathrm{T}}\cdot(f_{jh_b}\boldsymbol{A}^{(v)}_{h_b}\boldsymbol{\theta}^{(v)}_{jh_b})$ can be explained as how the user's tag feature matches the movie's player feature. The more positive the value is, the more the user's tag matches the movie's player. If we further remove the personal factors of feature weights $\{f_{ih_a}, f_{jh_b}\}$ and topic proportions $\{\boldsymbol{\theta}^{(u)}_{ih_a}, \boldsymbol{\theta}^{(v)}_{jh_b}\}$, the transformation matrices $\boldsymbol{A}^{(u)}_{h_a}$ and $\boldsymbol{A}^{(v)}_{h_b}$ can directly reflect how the t_a-th topic of user tags matches the t_b-th topic of movie actors, by calculating $(\boldsymbol{A}^{(u)}_{h_a}\tilde{\boldsymbol{\theta}}^{(u)}_{t_a})^{\mathrm{T}}(\boldsymbol{A}^{(v)}_{h_b}\tilde{\boldsymbol{\theta}}^{(v)}_{t_b})$. $\tilde{\boldsymbol{\theta}}^{(u)}_{t_a}$ is a $T^{(u)}_{h_a}$-dimensional topic proportion vector, with a value of 1 at the t_a-th dimension, and a value of 0 at the other dimensions. It denotes the t_a-th topic of user tags. Similarly, $\tilde{\boldsymbol{\theta}}^{(v)}_{t_b}$ is a $T^{(v)}_{h_b}$-dimensional topic proportion vector, denoting the t_b-th topic of movie players. The larger this value is, the stronger these two topics match. Thus by comparing the match strength between the two arbitrary topics, the model can answer questions like "What kind of movie players do "rebellious" users prefer?"

In a similar manner, the second term and the third term can be explained as how well individual users/items match topics of features, which corresponds to the second subtask in the recommendation interpretation problem. We still utilize the user tag feature and the movie player feature for illustrations. The matching strength between the i-th individual user and the t_b-th topic of the movie player, can be calculated by $(\boldsymbol{\varepsilon}^{(u)}_i)^{\mathrm{T}}(\boldsymbol{A}^{(v)}_{h_b}\tilde{\boldsymbol{\theta}}^{(v)}_{t_b})$. Similarly, the matching strength of the j-th individual movie and the t_a-th topic of the user tag, can be calculated by $(\boldsymbol{\varepsilon}^{(v)}_j)^{\mathrm{T}}(\boldsymbol{A}^{(u)}_{h}\boldsymbol{\theta}^{(u)}_{ih})$. By comparing the match strengths between an individual user/item and all topics of a certain feature, the model can answer the questions like "What kind of movie players does Tom prefer?"

Finally, for a user-item pair, the rating prediction is a linear combination of matching strengths between different kinds of information. By ranking these strength values, the model can show the impact of different features in generating a rating. The top positive ones are the main reasons why the user likes the item; and the top negative ones are the main factors why the user dislikes the item. Thus the model can answer questions like "When a user gives a high rating to a movie, is it because he likes those sorts of actors or those sorts of directors?"

4.3 Discussions

The proposed model can be seen as generalizations of many previous studies. If we remove all the topic models for user feature presentation, and retain only one PLSI model for item feature presentation, the model is simplified to the CTR model[11]. If we further remove the personal bias vector of $\boldsymbol{\varepsilon}^{(v)}_j$, the model is approximately simplified to the fLDA model[9]. The main advantage of our model compared with these previous studies is that we utilize multiple topic models to present multiple kinds of features. In this manner, the matching strengths among different feature topics can be modeled, for improving both interpretation and rating predictions. Our model is a joint model of matrix factorization and topic analysis. If we deal with them separately, which means learning the topic first and then doing a matrix factorization according to the topic analysis result, the model is simplified to the winner algorithm of KDD Cup 2012[1]. The advantage of jointly modeling is that the topic analysis can consider the "preference behavior" besides "looking", which is a kind of supervised topic modeling[37]. If we remove all features, our model will be simplified to the collective matrix factorization[34], where two rating matrices are factorized together.

5 Parameter Estimation

5.1 Maximum a Posterior

The parameters to be estimated include $\{\widehat{\boldsymbol{q}}, \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}\}$, which are detailed as,

$$\begin{aligned}
\widehat{\boldsymbol{q}} &= \{\boldsymbol{q}^{(u)}_i, \boldsymbol{q}^{(v)}_j, \boldsymbol{q}^{(s)}_k : 1 \leqslant i \leqslant N^{(u)},\\
&\quad 1 \leqslant j \leqslant N^{(v)}, 1 \leqslant k \leqslant N^{(s)}\},\\
\widehat{\boldsymbol{f}} &= \{\boldsymbol{f}^{(u)}_i, \boldsymbol{f}^{(v)}_j, \boldsymbol{f}^{(s)}_k : 1 \leqslant i \leqslant N^{(u)},\\
&\quad 1 \leqslant j \leqslant N^{(v)}, 1 \leqslant k \leqslant N^{(s)}\},\\
\widehat{\boldsymbol{A}} &= \{\boldsymbol{A}^{(u)}_{h_u}, \boldsymbol{A}^{(v)}_{h_v}, \boldsymbol{A}^{(s)}_{h_s} : 1 \leqslant h_u \leqslant M^{(u)},\\
&\quad 1 \leqslant h_v \leqslant M^{(v)}, 1 \leqslant h_s \leqslant M^{(s)}\},\\
\widehat{\boldsymbol{\theta}} &= \{\boldsymbol{\theta}^{(u)}_{ih_u}, \boldsymbol{\theta}^{(v)}_{jh_v}, \boldsymbol{\theta}^{(s)}_{kh_s} : 1 \leqslant i \leqslant N^{(u)},\\
&\quad 1 \leqslant j \leqslant N^{(v)}, 1 \leqslant k \leqslant N^{(s)}, 1 \leqslant h_u \leqslant M^{(u)},\\
&\quad 1 \leqslant h_v \leqslant M^{(v)}, 1 \leqslant h_s \leqslant M^{(s)}\},\\
\widehat{\boldsymbol{\varphi}} &= \{\boldsymbol{\varphi}^{(u)}_{h_u z_u}, \boldsymbol{\varphi}^{(v)}_{h_v z_v}, \boldsymbol{\varphi}^{(s)}_{h_s z_s} : 1 \leqslant h_u \leqslant M^{(u)},\\
&\quad 1 \leqslant h_v \leqslant M^{(v)}, 1 \leqslant h_s \leqslant M^{(s)}, 1 \leqslant z_u \leqslant T^{(u)}_{h_u},\\
&\quad 1 \leqslant z_v \leqslant T^{(v)}_{h_v}, 1 \leqslant z_s \leqslant T^{(s)}_{h_s}\}.
\end{aligned}$$

The pre-defined hyper-parameters are $\{\widehat{\boldsymbol{\lambda}}, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}\}$, detailed as,

$$\begin{aligned}
\widehat{\boldsymbol{\lambda}} &= \{\lambda^{(u)}, \lambda^{(v)}, \lambda^{(s)}, \lambda^{(f)}, \lambda^{(\boldsymbol{A})}\},\\
\widehat{\boldsymbol{\alpha}} &= \{\boldsymbol{\alpha}^{(u)}_{h_u}, \boldsymbol{\alpha}^{(v)}_{h_v}, \boldsymbol{\alpha}^{(s)}_{h_s} : 1 \leqslant h_u \leqslant M^{(u)},\\
&\qquad 1 \leqslant h_v \leqslant M^{(v)}, 1 \leqslant h_s \leqslant M^{(s)}\},\\
\widehat{\boldsymbol{\beta}} &= \{\boldsymbol{\beta}^{(u)}_{h_u}, \boldsymbol{\beta}^{(v)}_{h_v}, \boldsymbol{\beta}^{(s)}_{h_s} : 1 \leqslant h_u \leqslant M^{(u)},\\
&\qquad 1 \leqslant h_v \leqslant M^{(v)}, 1 \leqslant h_s \leqslant M^{(s)}\}.
\end{aligned}$$

The fixed parameters are $\{\gamma^{(uv)}, \gamma^{(us)}\}$.

We utilize the maximum a posterior (MAP) method to optimize the parameters. The likelihood of the proposed model is as follows.

$$\begin{aligned}
& p(\widehat{\boldsymbol{q}}, \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}} | \widehat{\boldsymbol{w}}, \widehat{\boldsymbol{r}})\\
\propto\ & p(\widehat{\boldsymbol{w}}, \widehat{\boldsymbol{r}} | \widehat{\boldsymbol{q}}, \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}) \times\\
& p(\widehat{\boldsymbol{q}}, \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}} | \widehat{\boldsymbol{\lambda}}, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}})\\
=\ & p(\widehat{\boldsymbol{r}} | \widehat{\boldsymbol{q}}) \times p(\widehat{\boldsymbol{w}} | \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}) \times\\
& p(\widehat{\boldsymbol{q}} | \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\lambda}}) \times p(\widehat{\boldsymbol{A}} | \widehat{\boldsymbol{\lambda}}) \times p(\widehat{\boldsymbol{f}} | \widehat{\boldsymbol{\lambda}}) \times\\
& p(\widehat{\boldsymbol{\theta}} | \widehat{\boldsymbol{\alpha}}) \times p(\widehat{\boldsymbol{\varphi}} | \widehat{\boldsymbol{\beta}}),
\end{aligned}$$

$$\begin{aligned}
p(\widehat{\boldsymbol{r}} | \widehat{\boldsymbol{q}}) = & \prod_{i=1}^{N^{(u)}} \prod_{j=1}^{N^{(v)}} \left(\mathcal{N}(r^{(uv)}_{ij} | (\boldsymbol{q}^{(u)}_i)^{\mathrm{T}} \boldsymbol{q}^{(v)}_j, \frac{1}{\lambda^{(uv)}} \boldsymbol{I}) \right)^{I^{(uv)}_{ij}}\\
& \prod_{i=1}^{N^{(u)}} \prod_{k=1}^{N^{(s)}} \left(\mathcal{N}(r^{(us)}_{ik} | (\boldsymbol{q}^{(u)}_i)^{\mathrm{T}} \boldsymbol{q}^{(s)}_k, \frac{1}{\lambda^{(us)}} \boldsymbol{I}) \right)^{I^{(us)}_{ik}},\\
p(\widehat{\boldsymbol{w}} | \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}) = & \prod_{\rho} \prod_{h=1}^{M^{(\rho)}} \prod_{i=1}^{N^{(\rho)}} \prod_{l=1}^{L^{(\rho)}_{ih}} \left(\sum_{z=1}^{T^{(\rho)}_h} \theta^{(\rho)}_{ihz} \varphi^{(\rho)}_{hzw_{ihl}} \right),\\
p(\widehat{\boldsymbol{q}} | \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\lambda}}) = & \prod_{\rho} \prod_{i=1}^{N^{(\rho)}}\\
& \mathcal{N}\left(\boldsymbol{q}^{(\rho)}_i \middle| \sum_{h=1}^{M^{(\rho)}} f^{(\rho)}_{ih} \boldsymbol{A}^{(\rho)}_h \boldsymbol{\theta}^{(\rho)}_{ih}, \frac{1}{\lambda^{(\rho)}} \boldsymbol{I} \right),\\
p(\widehat{\boldsymbol{f}} | \widehat{\boldsymbol{\lambda}}) = & \prod_{\rho} \prod_{i=1}^{N^{(\rho)}} \mathcal{N}\left(\boldsymbol{f}^{(\rho)}_i | 0, \frac{1}{\lambda^{(f)}} \boldsymbol{I} \right),\\
p(\widehat{\boldsymbol{A}} | \widehat{\boldsymbol{\lambda}}) = & \prod_{\rho} \prod_{i=1}^{N^{(\rho)}} \prod_{x} \prod_{y} \mathcal{N}\left(A^{(\rho)}_{xy} | 0, \frac{1}{\lambda^{(\boldsymbol{A})}} \right).
\end{aligned}$$

In optimizing likelihood, we employ the coordinate ascent method to update each kind of parameter respectively, by referring to the optimization procedure in [11]. In this method, when updating one parameter, all other parameters are fixed. By setting the derivative of the likelihood with respect to the active parameter to zero, the parameters of $\{\widehat{\boldsymbol{q}}, \widehat{\boldsymbol{f}}, \widehat{\boldsymbol{A}}\}$ can be optimized as follows.

$$\begin{aligned}
\boldsymbol{q}^{(u)}_i = & (\lambda^{(uv)} (\boldsymbol{Q}^{(v)})^{\mathrm{T}} \boldsymbol{C}^{(uv)}_{u_i} \boldsymbol{Q}^{(v)} +\\
& \lambda^{(us)} (\boldsymbol{Q}^{(s)})^{\mathrm{T}} \boldsymbol{C}^{(us)}_{u_i} \boldsymbol{Q}^{(s)} + \lambda^{(u)} \boldsymbol{I})^{-1}\\
& \Big(\lambda^{(uv)} (\boldsymbol{Q}^{(v)})^{\mathrm{T}} \boldsymbol{C}^{(uv)}_{u_i} \boldsymbol{r}^{(uv)}_i + \lambda^{(us)} (\boldsymbol{Q}^{(s)})^{\mathrm{T}}\\
& \boldsymbol{C}^{(us)}_{u_i} \boldsymbol{r}^{(us)}_i + \lambda^{(u)} \sum_{h=1}^{M^{(u)}} f^{(u)}_{ih} \boldsymbol{A}^{(u)}_h \boldsymbol{\theta}^{(u)}_{ih} \Big),\\
\boldsymbol{q}^{(v)}_j = & (\lambda^{(uv)} (\boldsymbol{Q}^{(u)})^{\mathrm{T}} \boldsymbol{C}^{(uv)}_{v_j} \boldsymbol{Q}^{(u)} + \lambda^{(v)} \boldsymbol{I})^{-1}\\
& \Big(\lambda^{(uv)} (\boldsymbol{Q}^{(u)})^{\mathrm{T}} \boldsymbol{C}^{(uv)}_{v_j} \boldsymbol{r}^{(vu)}_j +\\
& \lambda^{(v)} \sum_{h=1}^{M^{(v)}} f^{(v)}_{jh} \boldsymbol{A}^{(v)}_h \boldsymbol{\theta}^{(v)}_{jh} \Big),\\
\boldsymbol{q}^{(s)}_k = & (\lambda^{(us)} (\boldsymbol{Q}^{(u)})^{\mathrm{T}} \boldsymbol{C}^{(us)}_{s_k} \boldsymbol{Q}^{(u)} + \lambda^{(s)} \boldsymbol{I})^{-1}\\
& \Big(\lambda^{(us)} (\boldsymbol{Q}^{(u)})^{\mathrm{T}} \boldsymbol{C}^{(us)}_{s_k} \boldsymbol{r}^{(su)}_k +\\
& \lambda^{(s)} \sum_{h=1}^{M^{(s)}} f^{(s)}_{kh} \boldsymbol{A}^{(s)}_h \boldsymbol{\theta}^{(s)}_{kh} \Big),\\
f^{(\rho)}_{ih} = & (\lambda^{(\rho)} (\boldsymbol{q}^{(\rho)}_i)^{\mathrm{T}} \boldsymbol{A}^{(\rho)}_h \boldsymbol{\theta}^{(\rho)}_{ih} -\\
& \lambda^{(\rho)} \Big(\sum_{h'=1, h' \neq h}^{M^{(\rho)}} f^{(\rho)}_{ih'} \boldsymbol{A}^{(\rho)}_{h'} \boldsymbol{\theta}^{(\rho)}_{ih'} \Big)^{\mathrm{T}} \boldsymbol{A}^{(\rho)}_h \boldsymbol{\theta}^{(\rho)}_{ih}) /\\
& (\lambda^{(\rho)} (\boldsymbol{A}^{(\rho)}_h \boldsymbol{\theta}^{(\rho)}_{ih})^{\mathrm{T}} (\boldsymbol{A}^{(\rho)}_h \boldsymbol{\theta}^{(\rho)}_{ih}) + \lambda^{(f)}),\\
\boldsymbol{A}^{(\rho)}_h = & \Big(\sum_{i=1}^{N^{(\rho)}} \lambda^{(\rho)} (\boldsymbol{q}^{(\rho)}_i - \sum_{h'=1, h' \neq h}^{M^{(\rho)}} f^{(\rho)}_{ih'} \boldsymbol{A}^{(\rho)}_{h'} \boldsymbol{\theta}^{(\rho)}_{ih'})\\
& f^{(\rho)}_{ih} (\boldsymbol{\theta}^{(\rho)}_{ih})^{\mathrm{T}} \Big)\\
& \Big(\sum_{i=1}^{N^{(\rho)}} \lambda^{(\rho)} f^{(\rho)}_{ih} \boldsymbol{\theta}^{(\rho)}_{ih} f^{(\rho)}_{ih} (\boldsymbol{\theta}^{(\rho)}_{ih})^{\mathrm{T}} + \lambda^{(\boldsymbol{A})} \boldsymbol{I} \Big)^{-1}.
\end{aligned}$$

$\boldsymbol{r}^{(uv)}_i$ is an $N^{(v)} \times 1$ vector denoting the ratings of user i to items in site v; $\boldsymbol{r}^{(us)}_i$ is an $N^{(s)} \times 1$ vector denoting the ratings of user i to items in site s; $\boldsymbol{r}^{(vu)}_j$ is an $N^{(u)} \times 1$ vector denoting the ratings of item j from all users; and $\boldsymbol{r}^{(su)}_k$ is an $N^{(u)} \times 1$ vector denoting the ratings of item k from all users. A zero value is set when the rating is missing. $\boldsymbol{C}^{(uv)}_{u_i}$ is a diagonal $N^{(v)} \times N^{(v)}$

matrix with the j-th diagonal element being $I_{ij}^{(uv)}$; and $\boldsymbol{C}_{u_i}^{(us)}$ is a diagonal $N^{(s)} \times N^{(s)}$ matrix with the k-th element being $I_{ik}^{(us)}$. I is the indicator function to denote whether a rating is existing.

Parameters $\{\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}\}$ are probabilities. Thus we need to guarantee their values to be positive, and the sum of the probabilities of all values for a variable to be 1. We employ the projection gradient[39] method to solve the issue in the optimization procedure. The gradient of the likelihood with respect to $\{\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}\}$ is as follows. Please refer to [39] for details of the projection gradient algorithm.

$$p_{ihl}^{(\rho)}(z) \propto \theta_{ihz}^{(\rho)} \varphi_{hzw_{ihl}}^{(\rho)},$$

$$\varphi_{hzw}^{(\rho)} \propto \beta_{hw}^{(\rho)} + \sum_{i=1}^{N^{(\rho)}} \sum_{l=1}^{L_{ih}^{(\rho)}} I_l(w) p_{ihl}^{(\rho)}(z),$$

$$\frac{\partial \mathcal{L}}{\partial \theta_{ihz}^{(\rho)}} = \left(\alpha_{hz}^{(\rho)} + \sum_{l=1}^{L_{ih}^{(\rho)}} p_{ihl}^{(\rho)}(z) \frac{1}{\theta_{ihz}^{(\rho)}} - \lambda^{(\rho)} (\boldsymbol{q}_i^{(\rho)} - \sum_{h'=1}^{M^{(\rho)}} f_{ih'}^{(\rho)} \boldsymbol{A}_{h'}^{(\rho)} \boldsymbol{\theta}_{ih'}^{(\rho)} \right)^{\mathrm{T}} \boldsymbol{A}_{hz}^{(\rho)} (-f_{ih}^{(\rho)}).$$

5.2 Complexity Analysis

We suppose the dimensions of all latent vectors or topic proportions are equal to K; the word lengths of each user/item for all features are equal to L; R is the total number of ratings, and the vocabulary sizes for all features are equal to W. For each iteration, the computational complexity to update $\widehat{\boldsymbol{q}}$ is $O(RK^2 + K^3)$; the computational complexity to update $\widehat{\boldsymbol{f}}$ is $O((N^{(u)} + N^{(v)} + N^{(s)})K^2)$; the computational complexity to update $\widehat{\boldsymbol{A}}$ is $O((M^{(u)} + M^{(v)} + M^{(s)})K^3 + ((N^{(u)} + N^{(v)} + N^{(s)})K^2))$; and the computational complexity to update $\{\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}\}$ is $O((N^{(u)}M^{(u)} + N^{(v)}M^{(v)} + N^{(s)}M^{(s)})K^2 + (N^{(u)} + N^{(v)} + N^{(s)})WK)$. Although K^3 has occurred in the estimation of $\widehat{\boldsymbol{q}}$ and $\widehat{\boldsymbol{A}}$ due to the inverse calculation of the $K \times K$ matrix, this calculation is quite trackable in real cases. Because in each iteration, such calculations happen only $(2 + M^{(u)} + M^{(v)} + M^{(s)})$ times, and the dimension size K is not large either. Suppose we have incorporated 50 kinds of features, and the dimension K is set to 100, which has already been much larger than the real case. The calculation number of adding and multiplying is of the level of 50 M, which can be finished in less than 0.1 s by a CPU of 3 GHz. From the above analysis, it can be concluded that the complexity of the proposed model scales linearly with respect to the number of observations. By empirical study, the proposed algorithm can converge after around 10 iterations. Therefore, the proposed approach is very efficient, and can be utilized in very large datasets.

6 Experiments

In this section, we conduct several experiments to verify the performance of the proposed MFHLT model, for tasks of rating prediction and recommendation interpretation.

Our experiments aim to address the following questions.

1) To what extent can cross-site ratings improve active rating prediction performance? To what extent can cross-site content features improve the active rating prediction performance?

2) How does our approach perform compared with state-of-the-art algorithms for rating prediction?

3) How does our approach perform for recommendation interpretation?

To answer the first two questions, we employ competitive recommendation algorithms, together with the proposed MFHLT model, to verify performances after incorporating cross-site ratings and content features. Detailed studies are conducted to demonstrate the effectiveness of incorporating cross-site ratings and cross-site content features, respectively. To answer the last question, we conduct a qualitative study to demonstrate that the matchings among topics of features and individual users/items, which are learned by the proposed model, satisfy common standards. We also conduct a quantitative study to demonstrate that the MFHLT model can tell the major rationale of generating a rating.

6.1 Dataset

The dataset utilized in this paper is collected from three popular social media sites in China, including Douban①, Dianping②, and Sina Weibo③. Douban is a review system for books, movies and music; Dianping is a review system mainly for Chinese restaurants;

①http://www.douban.com, Apr. 2015.

②http://www.dianping.com, Apr. 2015.

③http://www.weibo.com, Apr. 2015.

and Sina Weibo is a micro-blog system. All the above social media sites have a large number of active users in China, which enables us to collect enough users who simultaneously have accounts in all of them. Thus they are ideal sources for analyzing cross-site user behavior.

We have collected two rating matrices, including the user-movie rating matrix from Douban and the user-restaurant rating matrix from Dianping. In addition, multiple content features are also collected. Let "C" denote the category feature and "W" denote the word-like feature. In Douban, we collect "movie tags (W)", "movie players (W)", "movie genres (W)", and "movie countries (C)" for movies; and we collect the "followees (W)" of each user, which is a kind of social information. In Dianping, we have collected "restaurant tags (W)", "restaurant comments (W)", and "restaurant prices (C)" for restaurants. In Sina Weibo, "ages&genders (C)" and "user tags" for users are collected.

The most difficult part of constructing the dataset is how to link users in different sites to guarantee that the three accounts are the same user, which is a very expensive process. We employ two algorithms[40-41] for this issue. In Liu *et al.*'s algorithm[40], the linking is based on how rare a username is. When a username is very rare, e.g., "pennystar88", and it occurs in the three sites, they are very likely to be the same user. Yuan *et al*'s algorithm[41] is more straightforward. Some users of one site might announce his/her accounts in other sites directly. Thus they design an algorithm to find these links automatically. Since both algorithms aim to link users across different sites with high precision (they cannot guarantee high recall), we can only find a limited number of linked users. Originally, 30 973 users are collected by matching users in Douban and Dianping. These users have rated 29 257 movies and 58 499 restaurants in total. However, a majority of these users only have ratings on one site. We remove these users, and only 1 285 users are left. To make the rating matrix denser for evaluation purpose, we also guarantee that each movie has at least three ratings, each restaurant has at least two ratings, and each user has at least one rating on both sites. Finally, 1 007 users, 4 537 movies, and 2 727 restaurants are collected in the dataset. There is a total of 184 608 ratings in the user-movie matrix, and 9 368 ratings in the user-restaurant matrix. Detailed statistics of these rating matrices are shown in Table 6. After this procedure, we further match the users with Weibo's user accounts. Six hundred and eighty-five of the users are linked to their Weibo accounts and their demographic features can be collected. The number of users here is small, but it is not surprising since linking users is a difficult task. Those linked users are indispensable resources for analyzing cross-site recommendation.

Table 6. Statistics of Rating Matrices

Statistics	Min. # Ratings	Max. # Ratings	Avg. # Ratings
User-movie	1	1 678	183.32
User-restaurant	1	285	9.30
Movie	3	489	40.69
Restaurant	2	47	3.43

6.2 Rating Prediction Performance

6.2.1 Experimental Setup

We utilize two metrics, the mean absolute error (MAE), and the root mean square error (RMSE), to evaluate rating prediction performance. Detailed definitions of these two metrics can be found in [17]. Both the two metrics measure errors. Thus a smaller MAE or RMSE value indicates a better performance. We randomly select 90% and 80% ratings of the dataset as training data, and the remained ratings as testing data. The selection process is carried out five times independently. In the testing data, we select the top 20% of users who have the least ratings in the training set. We denote this group of users as the "sparse" group. The original users are set as the "dense" group. We show the performances in these two sets respectively. In configuring the proposed model, we set $\lambda^{(u)} = \lambda^{(v)} = \lambda^{(s)} = 5.0$, $\lambda^{(f)} = \lambda^{(\boldsymbol{A})} = 2.0$, and $\gamma^{(uv)} = \gamma^{(us)} = 1.0$. We set each dimension of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to be 1.0. The length of each latent vector in matrix factorization is 30. The topic numbers of the content features are from 10 to 20.

6.2.2 Overall Performance

We compare the proposed model with the following previous methods.

1) PMF[3]: a typical competitive matrix factorization based algorithm.

2) CMF[34]: in this method, two rating matrices are jointly factorized.

3) ImSoc[42]: in this method, implicit social relations are utilized for improvements.

4) ExSoc[29]: in this method, explicit social relations are utilized for improvements.

5) SMF[43]: in this method, explicit social relations are utilized in a propagation manner to improve rating predictions.

6) FLDA[9]: in this method, the latent vectors of items are presented by the topic proportions learned from topic analysis from item features.

7) CTM[11]: in this method, an item's latent vector is presented by a linear combination of its topic proportion vector and its personal bias vector.

8) FMF[1]: in this method, topic analysis and matrix factorization are optimized in a separate manner. This method won the first place in the 2012KDD Cup④.

In our dataset, we can utilize Douban's data to help Dianping's data prediction, or Dianping's data to help Douban's data prediction. We choose the former one in showing overall performance because Dianping's data is sparser than Douban's, which makes the prediction more challenging. In the PMF model, only ratings from Dianping are utilized. In the CMF model, both ratings are utilized. In ExSoc & SMF, users' "followed" information is utilized as the explicit social information. In FLDA, CTR, and FMF, all content features are incorporated. But in the FLDA model and the CTM model, the vocabularies of all kinds of features are combined together to be presented by a single topic model.

Table 7 shows the overall performance. Fig.3 shows the performance when users are divided into different groups according to their number of ratings in the training set. It can be observed that the proposed model consistently outperforms other methods in all configurations. This demonstrates that the proposed approach is more effective in utilizing content features. By effectively incorporating the cross-site ratings and the cross-site content features in the sparse case, the MAE can be improved by 4.55%, and the RMSE can be improved by around 4.23%.

Table 7. Overall Performance Comparisons

Training Set	Metrics	PMF	CMF	ImSoc	ExSoc	SMF	FLDA	CTM	FMF	MFHLT
Dense 90%	MAE	0.875 9	0.852 8	0.848 6	0.847 4	0.846 8	0.843 7	0.842 1	0.840 3	**0.834 1**
	RMSE	1.206 5	1.183 8	1.181 9	1.181 2	1.180 2	1.178 1	1.175 8	1.171 8	**1.160 2**
Dense 80%	MAE	0.883 4	0.870 1	0.866 5	0.865 0	0.863 2	0.860 9	0.859 1	0.853 2	**0.843 9**
	RMSE	1.219 7	1.199 2	1.195 2	1.194 3	1.193 1	1.190 3	1.189 3	1.188 4	**1.183 7**
Sparse 90%	MAE	0.928 8	0.903 4	0.901 5	0.899 4	0.897 1	0.891 5	0.891 2	0.890 4	**0.886 5**
	RMSE	1.266 3	1.244 6	1.240 3	1.239 1	1.238 5	1.236 6	1.233 4	1.227 3	**1.212 7**
Sparse 80%	MAE	0.928 9	0.919 7	0.917 0	0.913 4	0.913 0	0.910 7	0.908 8	0.903 8	**0.891 7**
	RMSE	1.278 7	1.249 1	1.248 1	1.247 4	1.247 6	1.246 5	1.246 3	1.240 1	**1.231 7**

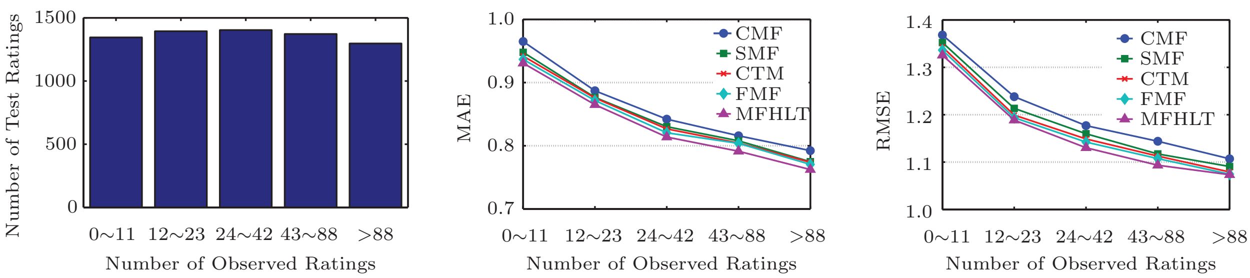


Fig. 3. Performance comparison on different users.

6.2.3 *Impact of Cross-Site Ratings*

To study the impact of cross-site ratings, we make comparisons between in-site ratings and cross-site ratings. We remove all content features from the proposed model. We randomly keep 50% of ratings in the training set as the origin training set. Then the remaining 50% of data is divided into several bins randomly with equal size. In each bin, we further divide the ratings into movie ratings and restaurant ratings. Then we add restaurant rating bins and movie rating bins into the training set gradually and respectively, to see how the performance would be changed. Fig.4 shows the improvements of MAE and RMSE by adding restaurant ratings and movie ratings. It can be observed

④ https://www.kddcup2012.org, May 2015.

that both kinds of ratings improve the prediction performance. But in-site ratings are more effective than cross-site ratings.

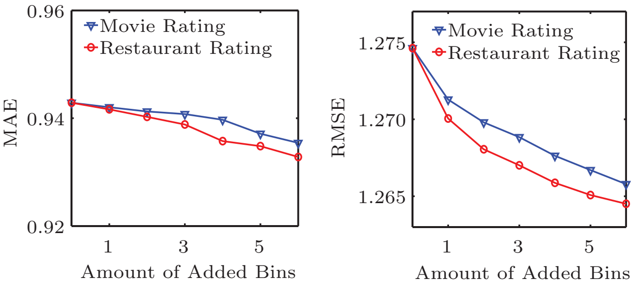


Fig.4. Comparisons between in-site ratings and cross-site ones.

6.2.4 *Impact of Cross-Site Features*

To study the impact of cross-site features, we conduct experiments to show the effectiveness of each feature in Table 8. Since only a single feature is incorporated, our model is simplified to the CTM model. From the table, it can be observed that each kind of cross-site feature is effective in improving recommendation results. The proposed model outperforms other algorithms consistently.

The advantage of our model is that the feature topics are learned according to the "user preference" from matrix factorization, as well as the co-occurrences of "words" in "documents". Table 9 shows the movie actor "Baoqiang Wang" and the actor topic he belongs to by independent topic analysis (FMF) and joint topic analysis (MFHLT). "Baoqiang Wang" is a popular comedy actor in China. By independent topic analysis, the topics were learned by referring to the co-occurrence relationship in the same movie. Thus the actors who have been in the same movie with "Baoqiang Wang" are clustered into this topic. Intuitively, most users like "Baoqiang Wang" because they like comedies. Thus they might not show interests in these co-occurred actors. By joint topic analysis, other comedy actors are clustered in the topic, which is more reasonable. This example directly reflects the advantage of joint modeling topic analysis and matrix factorization.

Table 8. Comparisons When Incorporating a Single Feature (90% as Training Data)

Approach	No Feature		User Tag		User Social		Rest. Comment		Restaurant Tag		Movie Actor		Movie Genre		Movie Tag	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
FLDA	0.8527	1.1839	0.8527	1.1803	0.8513	1.1826	0.8472	1.1792	0.8485	1.1796	0.8516	1.1820	0.8497	1.1802	0.8479	1.1798
FMF	0.8525	1.1835	0.8512	1.1797	0.8492	1.1813	0.8458	1.1734	0.8458	1.1786	0.8511	1.1803	0.8473	1.1758	0.8432	1.1745
MFHLT	**0.8525**	**1.1835**	**0.8473**	**1.1732**	**0.8469**	**1.1803**	**0.8416**	**1.1678**	**0.8414**	**1.1743**	**0.8482**	**1.1768**	**0.8436**	**1.1728**	**0.8392**	**1.1684**

Table 9. Joint vs Independent Topic Analysis

Method	Result
MFHLT	Lost on Journey (Comedy), Cars (Comedy), Crazy Stone (Comedy), The Simpsons (Comedy)
FMF	Fire of Conscience (Li, Liao, Wang), Better and Better (Wang, Tong), On the Edge (Li, Liao)

6.2.5 *Advantages of the Proposed Framework*

In rating predictions, most previous recommendations suffer from the difficulty in incorporating features in heterogeneous spaces. For example, the movie actor feature space and the user tag feature space are different. In solving this problem, the proposed framework has the following three advantages over previous models. First, the dimension of the feature space is reduced by topic models, making the features more descriptive. Take the actor feature for example. In the original space, there are hundreds of actors, with most actors being in one or two movies. In this case, it is difficult to identify similar actors. With the feature dimension reduction, similar actors are presented by similar latent vectors. This helps alleviate the data sparsity problem. Second, multiple kinds of features from heterogenous feature spaces are linearly combined naturally by the proposed mapping matrices. From the experimental results, it can be seen that this method is effective for combining features in heterogeneous spaces. Third, the feature dimension reduction and the mapping matrices are jointly optimized. In this way, these two tasks are co-learned to optimize the global objective. Considering the previous "Baoqiang Wang" example, the topic analysis will have a bias to optimize rating prediction. This is the key reason why our method can outperform the FMF method.

6.2.6 More Discussions

Parameters $\gamma^{(uv)}$ and $\gamma^{(us)}$ control the balance between the topic analysis optimization and the matrix factorization optimization. The larger these two values are, the more weight will be used for the matrix factorization objective. Fig.5 illustrates the sensitivity analysis of the two parameters. We also show the converging speed of the proposed model in Fig.6. It is observed that the model converges in around 10 iterations.

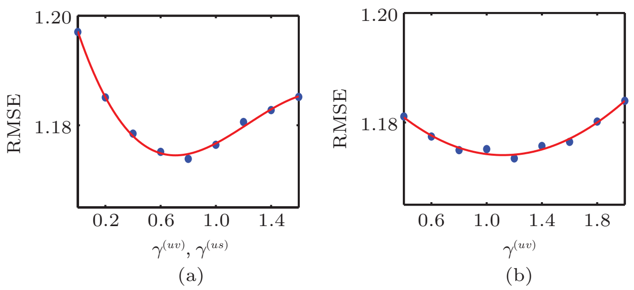


Fig.5. Sensitivity analysis for γ. (a) Two values are changed. (b) Only $\gamma^{(uv)}$ is changed.

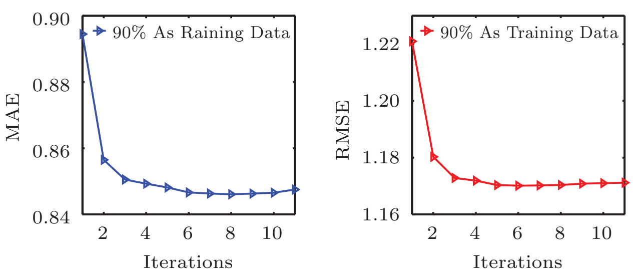


Fig.6. Converging speed of the proposed model.

6.3 Interpretation Performance

6.3.1 Qualitative Study

In the qualitative study, we demonstrate whether the learned matching relationships among features and individual users/items meet common standard. We illustrate two user tag feature topics, as shown in Table 10. We illustrate matched restaurant feature topics and matched movie feature topics. The matching is conducted by the calculations discussed in Subsection 4.2. From the table, it can be concluded that the matching relationships among feature topics satisfy our common standard. The first topic is related to cartoons. These users prefer fast and cheap restaurants and cartoon roles. The second topic is related to adult movies and TV. These users prefer more fashion and expensive food, and well-known fashion actors. In Table 11, we illustrate two individual users, with their review histories. From the calculation in Subsection 4.2, the first user matches the first user tag topic, and the second user matches the second user tag topic in Table 10. The matching relationships between features and individual users also satisfy our common standard. In Table 11, we illustrate the recommended items of the proposed model and the previous model. If the model can explain the rationale of the ratings well, the recommended items should be more reasonable. In this case, our model fits the user's intention better as it is recommending cartoons to a child. This example illustrates another reason why the proposed model has better performance. In fact, simultaneously solving the rating prediction task and the interpretation task can help each other in performances.

6.3.2 Quantitative Study

In the quantitative study, we utilize the following method to evaluate whether the proposed model can interpret the rationale of a rating accurately. For each user-item pair, besides the rating, some users also give tags to the item. We assume the tag can reflect the main reasons for generating the rating. For example, if a user gives a 5-star rating to the movie "Mission Impossible", and he/she gives a tag "action", in our assumption, this user likes this movie, because the "ac-

Table 10. Qualitative Study of Group Interpretation

User Group	User Group Tag	Restaurant	Movie
User group 1	Doraemon, Polar Bear, SpongeBob, Maruko, Dr. Slump, Humorous, Bosozoku	Tag: Deep-fried Dough Sticks, Tofu with Preserved Eggs, Rice Wine, Noodles Served with Such Sauce, Chili and Sour Potato Price Range: $\leqslant$ \$3	Actor: Kamiya Hiroshi, Kobayashi Sanae, Mitsuishi Kotono, Sawashiro Miyuki, Kiyokawa Motomu, Tachiki Fumihiko Genre: Cartoon, Story, Comedy Country: America
User group 2	American Film, US TV Series, Europe and America, Vampire, Fever	Tag: Filet Steak, Spicy Beef Ribs, Bacon, Chocolate, Oxtail Soup, Belgian Chocolate, Tuna Price Range: $\leqslant$ \$3	Actor: Milla Jovovich, Dustin Hoffman, Jean Reno, Harvey Keitel, Catherine Zeta-Jones, Arnold Schwarzenegger, Vincent Cassel Genre: Story, Fantasy, Adventure Country: Japan

Table 11. Qualitative Study of Individual Interpretation

User	User Tag	Review History	Recommended Items of MFHLT	Recommended Items of FMF
User 1	Doraemon	Chibi Maruko chan (5–star)	Mononoke (Cartoon)	Life is Beautiful (Story)
	SpongeBob	Dr. Slump (5–star)	Gintama (Cartoon)	Hotel Rwanda (Story)
	Dr. Slump	Rio (5–star)	Fruits Busket (Cartoon)	The Cove (Story)
	Comic	Triangle (1–star)	Code Geass Lelouch (Cartoon)	The School of Rock (Comedy)
	Leo	Deadly Delicious (2–star)	Yondemasu yo Azazel San (Cartoon)	Hyeong-Cheol Kang (Story)
User 2	American Film	Grey's Anatomy Season 6 (5–star)	The L Word Season 2 (US TV)	Mewtwo VS Mew (Cartoon)
	US TV Series	Castle Season 1 (5-star)	Iron Man 3 (US Film)	Bean (Comedy)
	Europe and America	Step Up (5-star)	Ugly Betty Season 2 (US TV)	Kang Xi Kingdom (Story)
	Harry Potter	Romance in the Rain (1–star)	Fringe Season 1 (US TV)	Desperate Housewives Season 4 (US TV)
	Avril	CJ7 (2–star)	Sex and a Half Men Season 1 (US TV)	Hotaru no Haka (Cartoon)

tion" genre feature attracts him/her. Our model can also calculate the weight of each feature in determining a rating as discussed in Subsection 4.2. The top-weighted features can be seen as the reasons for generating the active rating. Therefore, we compare the feature selected from our model and the original tag. If our feature can meet the tag, it can be concluded that our model is accurate in its interpretation. In the testing set of the user-movie matrix, we collect 5 475 tags for the corresponding ratings. In our method, we utilize the top word in the top-weighted feature topic as interpretations. We observe that most tags are related to the player, genre, or country of the movie, and thus we select the three pieces of information as three baseline predictors. If the word provided by each method is in the tag, we say this interpretation is correct. Table 12 shows the precisions of our method and the baselines. It can be concluded that our method outperforms the baselines significantly. This quantitative analysis supports that the proposed model can explain the rationale behind the ratings.

Table 12. Quantitative Study for Interpretation

Dataset	Our Method (%)	Player (%)	Genre (%)	Country (%)
Dense	64.03	21.45	39.58	34.28
Sparse	63.95	18.16	35.91	37.68

7 Conclusions

In this paper, we studied 1) how to utilize cross-site ratings and content features to improve the performances of recommender systems and 2) how to make the recommendation interpretable. We proposed a joint model of matrix factorization and topic analysis as the recommendation framework. Besides effectively and efficiently incorporating the cross-site information, the proposed model can interpret the rationale behind a rating by learning the matching strengths among content features and individual users/items. Through experimental verification in a real-world dataset, the proposed model is demonstrated to be effective in improving recommendation performance and providing interpretation for recommendation results. Meanwhile, it is also concluded that by incorporating cross-site ratings and content features, the recommendation performance does achieve significant improvements in both MAE (improved by 4.55%) and RMSE (improved by 4.23%).

A limitation of the current proposed framework is that the features incorporated are required to be bag-of-words features. This limitation comes from utilizing topic models to present latent feature vectors. In future work, the Gaussian mixture model could be utilized to present numerical features, and can be incorporated into the current framework. This will make the proposed model more generalizable.

References

[1] Chen T, Tang L, Liu Q *et al.* Combining factorization model and additive forest for collaborative followee recommendation. In *Proc. 2012 KDD Cup Workshop*, Aug. 2012.

[2] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer*, 2009, 42(8): 30–37.

[3] Salakhutdinov R, Mnih A. Probabilistic matrix factorization. In *Proc. the 21st Annual Conference on Neural Information Processing Systems*, Dec. 2007, pp.1257–1264.

[4] Töscher A, Jahrer M, Bell R M. The BigChaos solution to the Netflix grand prize. 2009. http://www.netflixprize.com/assets/GrandPrize2009_BPC_BigChaos.pdf, April 2015.

[5] Hu L, Cao J, Xu G, Cao L, Gu Z, Zhu C. Personalized recommendation via cross-domain triadic factorization. In

Proc. the 22nd International World Wide Web Conference, May 2013, pp.595–606.

[6] Agarwal D, Chen B C. Regression-based latent factor models. In *Proc. the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, June 28–July 1, 2009, pp.19–28.

[7] Xu Z, Tresp V, Rettinger A, Kersting K. Social network mining with nonparametric relational models. In *Proc. the 2nd Int. Conf. Advances in Social Network Mining and Analysis*, Aug. 2008, pp.77–96.

[8] Sharma A, Cosley D. Do social explanations work? Studying and modeling the effects of social explanations in recommender systems. In *Proc. the 22nd International World Wide Web Conference*, May 2013, pp.1133–1143.

[9] Agarwal D, Chen B C. fLDA: Matrix factorization through latent dirichlet allocation. In *Proc. the 3rd ACM International Conference on Web Search and Data Mining*, Feb. 2010, pp.91–100.

[10] McAuley J, Leskovec J. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proc. the 7th ACM Conference on Recommender Systems*, Oct. 2013, pp.165–172.

[11] Wang C, Blei D. Collaborative topic modeling for recommending scientific articles. In *Proc. the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2011, pp.448–456.

[12] Hofmann T. Probabilistic latent semantic indexing. In *Proc. the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 1999, pp.50–57.

[13] Breese J, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. the 14th Conference on Uncertainty in Artificial Intelligence*, July 1998, pp.43–52.

[14] Cai Y, Leung H F, Li Q, Min H, Tang J, Li J. Typicality-based collaborative filtering recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 2014, 26(3): 766–779.

[15] Deshpande M, Karypis G. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems*, 2004, 22(1): 143–177.

[16] Liu N N, Yang Q. Eigenrank: A ranking-oriented approach to collaborative filtering. In *Proc. the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2008, pp.83–90.

[17] Ma H, King I, Lyu M R. Effective missing data prediction for collaborative filtering. In *Proc. the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2007, pp.39–46.

[18] Hofmann T. Collaborative filtering via Gaussian probabilistic latent semantic analysis. In *Proc. the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 28-Aug. 1, 2003, pp.259–266.

[19] Si L, Jin R. Flexible mixture model for collaborative filtering. In *Proc. the 20th Annual International Conference on Machine Learning*, Aug. 2003, pp.704–711.

[20] Zhang Y, Koren J. Efficient Bayesian hierarchical user modeling for recommendation system. In *Proc. the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2007, pp.47–54.

[21] Porteous I, Bart E, Welling M. Multi-HDP: A non parametric Bayesian model for tensor factorization. In *Proc. the 23rd AAAI Conference on Artificial Intelligence*, July 2008, pp.1487–1490.

[22] Salakhutdinov R, Mnih A, Hinton G. Restricted Boltzmann Machines for collaborative filtering. In *Proc. the 24th Annual International Conference on Machine Learning*, June 2007, pp.791–798.

[23] Salakhutdinov R, Mnih A. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *Proc. the 25th Annual International Conference on Machine Learning*, July 2008, pp.880–887.

[24] Rennie J D, Srebro N. Fast maximum margin matrix factorization for collaborative prediction. In *Proc. the 22nd Annual International Conference on Machine Learning*, Aug. 2005, pp.713–719.

[25] Yu K, Zhu S, Lafferty J, Gong Y. Fast nonparametric matrix factorization for large-scale collaborative filtering. In *Proc. the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2009, pp.211–218.

[26] Ma H, Zhou T C, Lyu M R, King I. Improving recommender systems by incorporating social contextual information. *ACM Transactions on Information Systems*, 2011, 29(2): Article No. 9.

[27] Wu S, Sun J, Tang J. Patent partner recommendation in enterprise social networks. In *Proc. the 6th ACM International Conference on Web Search and Data Mining*, Feb. 2013, pp.43–52.

[28] Jamali M, Lakshmanan L. HeteroMF: Recommendation in heterogeneous information networks using context dependent factor models. In *Proc. the 22nd International World Wide Web Conference*, May 2013, pp.643–654.

[29] Ma H, King I, Lyu M R. Learning to recommend with social trust ensemble. In *Proc. the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2009, pp.203–210.

[30] Stern D H, Herbrich R, Graepel T. Matchbox: Large scale online Bayesian recommendations. In *Proc. the 18th International World Wide Web Conference*, Apr. 2009, pp.111–120.

[31] Zhang Y, Nie J. Probabilistic latent relational model for integrating heterogeneous information for recommendation. Technical Report, School of Engineering, University of California Santa Cruz, 2010.

[32] Shan H, Banerjee A. Generalized probabilistic matrix factorizations for collaborative filtering. In *Proc. the 10th IEEE International Conference on Data Mining*, Dec. 2010, pp.1025–1030.

[33] Ma H, Zhou D, Liu C, Lyu M R, King I. Recommender systems with social regularization. In *Proc. the 4th International Conference on Web Search and Data Mining*, Feb. 2011, pp.287–296.

[34] Singh A P, Gordon G J. Relational learning via collective matrix factorization. In *Proc. the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2008, pp.650–658.

[35] Zhang X, Cheng J, Yuan T, Niu B, Lu H. TopRec: Domain-specific recommendation through community topic mining in social network. In *Proc. the 22nd International Conference on World Wide Web*, May 2013, pp.1501–1510.

[36] Tang J, Wu S, Sun J, Su H. Cross-domain collaboration recommendation. In *Proc. the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2012, pp.1285–1293.

[37] Blei D M, McAuliffe J D. Supervised topic models. In *Proc. the 21st Annual Conference on Neural Information Processing Systems*, Dec. 2007, pp.121–128.

[38] Blei DM, Ng A Y, Jordan M I. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003, 3: 993–1022.

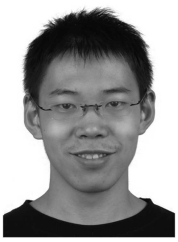
[39] Bertsekas D. Nonlinear Programming (2nd edition). Athena Scientific, 1999.

[40] Liu J, Zhang F, Song X, Song Y I, Lin C Y, Hon H W. What's in a name?: An unsupervised approach to link users across communities. In *Proc. the 6th ACM International Conference on Web Search and Data Mining*, Feb. 2013, pp.495–504.

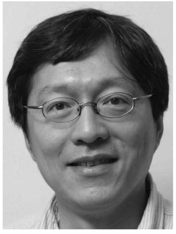
[41] Yuan N J, Zhang F, Lian D, Zheng K, Yu S, Xie X. We know how you live: Exploring the spectrum of urban lifestyles. In *Proc. the 1st Conference on Online Social Networks*, Oct. 2013, pp.3–14.

[42] Ma H, King I, Lyu M R. Learning to recommend with explicit and implicit social relations. *ACM Transactions on Intelligent Systems and Technology*, 2011, 2(3): Article No. 29.

[43] Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proc. the 4th ACM Conference on Recommender Systems*, Sept. 2010, pp.135–142.



Xin Xin is currently an assistant professor of the School of Computer Science, Beijing Institute of Technology. He received his B.S. and M.S. degrees from the Department of Computer Science and Technology, Tsinghua University, Beijing, in 2006 and 2008, respectively, and his Ph.D. degree in computer science from The Chinese University of Hong Kong in 2011. His research interests include data mining, machine learning, etc. He is a member of CCF, ACM, and IEEE.



Chin-Yew Lin received his Ph.D. degree in computer engineering from the University of Southern California in 1997. Currently, he is a principal researcher and principal research manager of the Knowledge Mining group at Microsoft Research. His research interests include automated summarization, information extraction, natural language processing, problem solving, knowledge mining, social computing, and machine learning. He is a member of ACM, IEEE, and ACL.

Xiao-Chi Wei received his B.S. degree in computer science and technology from Liaoning University, Shenyang, in 2012. Currently, he is a Ph.D. candidate in the School of Computer Science, Beijing Institute of Technology. His research interests include data mining, machine learning, natural language processing, etc. He is a member of CCF and ACM.

He-Yan Huang is currently a professor and the dean of the School of Computer Science in Beijing Institute of Technology. She received her Ph.D. degree in computer science and technology in 1989 from Institute of Computing Technology, Chinese Academy of Sciences, Beijing. Her current research interests include machine learning, information retrieval, and natural language processing. She is a member of ACM.