IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 8, 2017, accepted August 31, 2017, date of publication September 13, 2017, date of current version October 12, 2017.

*Digital Object Identifier 10.1109/ACCESS.2017.2751682*

# Semantic Preference-Based Personalized Recommendation on Heterogeneous Information Network

**LIANG HU[1], YU WANG[1], ZHENZHEN XIE[1], AND FENG WANG[1,2]**
[1]College of Computer Science and Technology, Jilin University, Changchun 130012, China
[2]Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China

Corresponding author: Feng Wang (wangfeng12@mails.jlu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61701190, in part by the National Key R&D Plan of China under Grant 2017YFA0604500, in part by the National Sci-Tech Support Plan of China under Grant 2014BAH02F00, in part by the Youth Science Foundation of Jilin Province of China under Grant 20160520011JH, and in part by the Youth Sci-Tech Innovation Leader and Team Project of Jilin Province of China under Grant 20170519017JH.

**ABSTRACT** In recent years, the Internet has become an indispensable part of people's lives, and it offers increasingly comprehensive information tailored to people's personal preferences as well as commodity attribute information. Consequently, many researchers have used external information to improve recommendation technology. However, most previous studies consider only adding single relationship types, such as social networking friend-relationships. In the real world, considering multiple types of external relations can more accurately determine the reason why a user selected an item. To address this problem, in this paper, we propose a hybrid method called the semantic preference-based personalized recommendation on heterogeneous information networks (SPR), which combines user feedback scores with heterogeneous information networks. This method can improve recommendation problems by considering multiple types of external relationships. To apply the method, we first introduce a similarity measure between users based on a user's potential preferences in the meta-path and design the recommended model at the global and individual level. Finally, we perform experiments on two real-world data sets, finding that the SPR method achieves better results compared with the several widely employed and the state-of-the-art recommendation methods.

**INDEX TERMS** Recommendation technology, external relationships, heterogeneous information network.

## I. INTRODUCTION

With the explosive development of the Internet and big data, users now have access to large amounts of optional information when shopping. To facilitate selecting the appropriate items, recommendation systems emerged and are now widely used.

Collaborative filtering methods [1] have been widely used in many recommendation applications. These methods utilize historical information concerning interactions between users and items recommended by similar users. However, in the real world, users usually interact with only a limited number of items. In the face of such cold start and sparse feedback data problems, collaborative filtering methods often do not work very well. Therefore, some recent studies add external knowledge to help the recommender system and alleviate these problems (e.g., the social information of users and the attribute information of the items) [2], [3]. Nevertheless, these studies are mostly based on a single type of additional information. To make better use of user information, Yu *et al.* [4] consider the combination of recommendation system and heterogeneous information network. They use the implicit feedback data to combine the additional information of items to improve the recommended performance. The method deals with implicit feedback data, but it increased the complexity of the recommended model undoubtedly since the method needs additional clustering algorithm to deal with the sparse issue of the single user feedback data.

In this paper, we study a recommendation method in a heterogeneous information network. Unlike previous studies, we utilize multiple types of additional entity information, combined with the explicit feedback data and user-item collaborative filtering to improve the recommended results. For movie recommendation, in addition to the interaction information between users and movies, we add information such as user preferences for actors, directors, and categories.

Previous studies have relied on personal ratings or user feedback data to estimate user preferences. Usually, we regard high ratings as an indication that other users like a movie and low ratings as evidence that the movie fails to satisfy people's expectations. Then we use the collaborative filtering method to determine similar merchandise recommendations among users. These methods have achieved good results. However, in the real world, movie recommendations–for example, a user choosing to watch a certain movie–typically have a motive. For example, perhaps the user likes a particular actor in a movie, a certain type of movie, or likes a movie because his friends watched the same movie and recommended it highly. These movie-related factors are all reasons why users choose to watch a movie; therefore, we have reason to consider such reasons as more representative of a users' preferences.

Our contributions in this research are summarized as follows:

1) We study the personalized recommendation method with rating data in heterogeneous information networks.
2) We utilize the meta-path in heterogeneous information networks to represent the preferences of the user and construct the user's preference model.
3) Our studies utilize two real-world datasets, Douban and Yelp, to demonstrate the performance of our method.

The rest of the paper is organized as follows. In Section 2, data processing, the concepts of the heterogeneous information network and the meta-path are introduced. We propose two models that address global and personalized levels in Section 3. The experiments are introduced in Section 4, and Section 5 gives a brief summary of related works. Finally, we conclude our work in Section 6.

## II. PRELIMINARIES AND PROBLEM DEFINITION

In this section, we first introduce binary user feedback and then introduce the concepts of heterogeneous information networks and meta-paths. Finally, we give a formal problem definition.

### A. USER FEEDBACK ANALYSIS

With m users, $U = \{u_1, \ldots, u_m\}$ and n items $I = \{e_1, \ldots, e_n\}$, we use a similar definition of binary user feedback to that of the definition of user implicit feedback [4]; we define the user feedback matrix $R \in \mathbb{R}^{m \times n}$ as follows:

$$R_{ij} = \begin{cases} 1 & \text{if } u_i \text{ has rated } e_j; \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The user feedback matrix R is constructed by the users' ratings of items. For example, if user $i$ rated item $j$, the $R_{i,j}$ value is 1 in $\mathbb{R}^{m \times n}$; otherwise the value is 0. In the user feedback matrix, the value 1 means only that users have rated the items–for example, a user may have watched a movie or used a business service and then rated it. Regardless of whether the rating is high or low, the user chose to watch the movie because he or she was presumably interested in that movie. A user might dislike the movie after watching it and give it a low rating. A business rating is based on the same reasoning. Similarly, the value 0 does not mean that users dislike the items; it only indicates that the users have not rated in those items.

The above definition of the user feedback matrix is used mainly to convert users' rating scores of items into a binary relationship between users and items; during this process, the rating value loses its meaning. A single score reflects the users' level of recognition for an item, while a large number of scores can also reflect the quality of the item itself. Therefore, we combine the users' rating data for an item to present an additional concept called score threshold.

*Definition 1:* Score threshold. A score threshold is defined as a value that divides the rating score of items into high-score items and general-score items. If the score of an item is greater than or equal to the threshold value, the item is a high-score item; otherwise, it is a general-score item.

For example, in the Douban dataset, we can first calculate the score of each movie based on user data for different movie scores and then compare the movie scores with the threshold score; if the score of the movie is higher than the threshold score, the movie is a high-score movie.

### B. HETEROGENEOUS INFORMATION NETWORK-BASED RECOMMENDATION

Similar to [5], we define a heterogeneous information network as follows:

*Definition 2:* Heterogeneous Information Network. A heterogeneous information network (HIN) is defined as a directed graph with multiple types of nodes or multiple types of links. It can be denoted as $G = (V, E)$, where $V$ is the set of nodes, and $E$ is the set of links. In addition, a heterogeneous information network should be associated with a node-type mapping function $\phi = V \rightarrow A$ and a link-type mapping function $\psi = E \rightarrow L$. Each node $v \in V$ belongs to a particular node type $\phi(v) \in A$, $A$ is a set consists of different types of entities, and each link $l \in E$ belongs to a particular link type $\psi(l) \in L$, $L$ is a set consists of different relations that between different types of entities.

The heterogeneous information network is an abstraction of the real world, and it focuses on entities and relations between entities. We use a network schema to represent the nodes and relation types in a heterogeneous information network, denoted by $T_G = (A, L)$. As shown in Figure 1, the Douban movie network and the Yelp network can be considered as a heterogeneous information network. In Figure 1(a), node type $A$ includes user, movie, director, actor, genre, and rating score, while the relation type set $L$ includes user-movie-director, user-movie-actor and so on.

*Definition 3:* Meta-Path. A meta path [6] P is a path defined on the graph of network schema $T_G = (A, L)$ and is denoted as $A_1 \xrightarrow{L_1} A_2 \xrightarrow{L_2} \ldots \xrightarrow{L_i} A_{i+1}$, $A_i \in A$, which defines a composite relation $L = L_1 \circ R_2 \circ \ldots \circ L_i$ between type $A_1$ and $A_{i+1}$, where $\circ$ denotes the composition operator on relations.

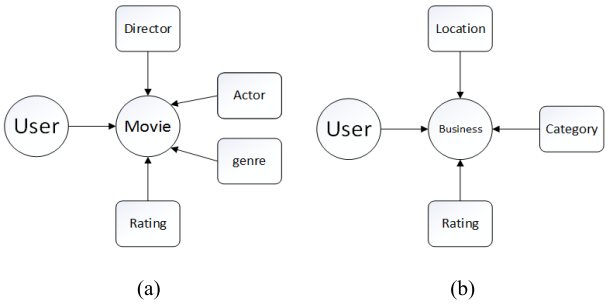


**FIGURE 1.** Network schema of a heterogeneous information network.

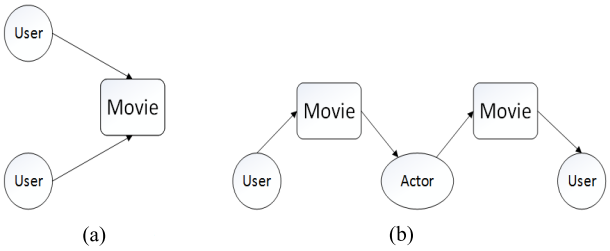


**FIGURE 2.** In a heterogeneous information network, the meta-path carries explicit semantic information.

Examples of meta-paths defined in the network schema in figure 1(a) include the''user-movie-actor'' path, the ''actor-movie-genre'' path and so on. The meta-path can be semantically explanatory, and when two nodes are connected via different paths, they hold different semantic information. In this paper, we choose only those meta-paths with explicit semantic information. For example, in Figure 2, the semantic information expressed by ''user-movie-actor'' path is that user $u$ watched a movie that actor $a$ starred in. In our proposed model, on one hand, we use the semantic information expressed by these meta-paths to provide an interpretation of the user's feedback behavior. On the other hand, we calculate the similarity of users based on the meta-paths.

### C. PROBLEM DEFINITION

In a heterogeneous information network, the meta-paths provide a variety of hierarchical semantic information that enables the representations of the user data and item data of the recommendation system and the complex relationships between them to be represented in an information network at the same time. Therefore, based on the HIN concept, we define the recommendation problem studied in this paper as follows:

*Definition 4:* Problem Definition. Given a heterogeneous information network $G$ with user feedback $R$, we aim to build a personalized recommendation model for user $u_i$ based on historical user feedback to predict rating score of user $u_i$ for unobserved interactions.

## III. SEMANTIC PATH-BASED RECOMMENDATION METHOD

We introduce a similarity method based on a combination of the meta-path and user feedback to analyze user' potential preferences for items. We then define a recommendation function based on the user's potential preferences. Finally, we introduce the personalized recommendation models at the end of this section.

### A. META-PATH-BASED SIMILARITY MEASURE METHOD

As described above, we divided the users' score data into two parts for analysis. On the one hand, we use the rating behavior to construct a binary relationship between users and items, which is used to represent the interactions between users and items; e.g., a user watched a movie or visited a restaurant. On the other hand, we use the numerical value of the score to filter the quality of an item–to determine the high-quality items–as a high-score item. For the interactions between users and items, we use the value 1 to represent a user's interest in an item. At the same time, by combining the semantic information expressed by the meta-paths, we can closely approximate the reason why a user might be interested in an item. After we understand the user's preferences, we can recommend items to that user based on other users with similar preferences.

After determining the reasoning of similar users, we need to know how to measure it. Here, we use a topological measuring method, PathSim, which was proposed in [6]:

*Definition 5:* PathSim: PathSim is a meta-path based similarity measure. Given a symmetric meta-path, the PathSim value between two objects of the same type x and y is calculated as follows:

$$s(x, y) = \frac{2 \times |\{p_{x\to y} : p_{x\to y} \in \mathcal{P}\}|}{|\{p_{x\to x} : p_{x\to x} \in \mathcal{P}\}| + |\{p_{y\to y} : p_{y\to y} \in \mathcal{P}\}|} \tag{2}$$

where $p_{x\to y}$ is a path instance between $x$ and $y$, $p_{x\to x}$ is a path instance between $x$ and $x$, and $p_{y\to y}$ is a path instance between $y$ and $y$.

The PathSim value of given meta-path P is defined regarding two parts: (1) their connectivity specified by the number of paths between them following $P$; and (2) the balance of their visibility, where the visibility is defined as the number of path instances between themselves. So the value of $s(x, y)$ is in the range of 0 to 1, the value is more close to 1, more similar. On the contrary, the value is more close to 0, more dissimilar. The time complexity for computing each path is about $O(d)$ on average and $O(m^2)$ in the worst case for all users and items, where $d$ is the non-zero element for users rated items, $m$ is the number of users.

In our model, we focus only on meta-paths in the format of user-item-*-item-user, because these paths have clear and meaningful semantic explanations. A meta-path represents a user's preference (the reason why the user chooses the item), and common better choices between users can reflect commonality between users. However, there is another case, and we give an example to illustrate it. Sometimes, a user chooses to watch a movie, not because of the actor or the director based on their preferences, but because the movie received a high rating score, as in the case of the high-score

item defined in Section 2. In this case, we filter the high-score item through the score threshold mentioned in Section 2 and construct the special meta-path user-item(high score)-user.

### B. RATING PREDICTION BY SIMILARITY

Through the similarity measure method above, we can find users similar to a target user under a given meta-path. Then, we can predict the target user's scores on items based on the rating scores of similar users. Assume $\widetilde{\mathcal{R}} \in \mathbb{R}^{m\times n}$ is the rating matrix, where $\widetilde{\mathcal{R}}_{m,n}$ denotes the rating score of $m$ users on $n$ items; and $S \in \mathbb{R}^{m\times m}$ is the user similarity matrix, in which $S^{(l)}_{u,v}$ denotes the similarity between user $u$ and user $v$ under the path $P_l$. We define the score prediction method as follows:

$$\overset{\sim}{\mathcal{R}}{}^{(l)}_{u,i} = \frac{\sum_{v\in U, v\neq u} S^{(l)}_{u,v} \cdot R_{v,i}}{\sum_{v\in U, v\neq u} S^{(l)}_{u,v}} \tag{3}$$

According to (3), we can compute the time complexity for each path, $S \in \mathbb{R}^{m\times m}$, $\widetilde{\mathcal{R}} \in \mathbb{R}^{m\times n}$, so the time complexity of the matrix multiplication is $O(m^3)$. We can obtain the predicted rating score of a user on an item under a given path. Then, the item with the highest score will be recommended to the target user.

### C. GLOBAL RECOMMENDATION MODEL

In the global recommendation model, n paths $P = \{P_1, \ldots, P_n\}$, $\overset{\sim}{\mathcal{R}}{}^{(l)}$ denotes the predicted rating matrix under the path $P_l$, which $P_l \in P$. The predicted rating scores indicate the possibility of certain user-item interactions under a certain meta-path semantic. We can obtain different predicted rating matrices under different semantic paths. In the global recommendation model, we combine the ratings from different paths and–considering that different semantic relations may have different levels of importance–define a global recommendation model as follows:

$$\overset{\sim}{\mathcal{R}}_{u,i} = \sum_{l=1}^{\mathcal{P}} \theta_l \cdot \overset{\sim}{R}{}^{(l)}_{u,i} \tag{4}$$

where $\theta_l$ is the weight for the path $P_l$. Based on the non-negative property of the features, we add $\theta_l \geq 0$ as a constraint.

To make the predicted scoring matrix and the true scoring matrix as similar as possible, we use the least squares method for parameter estimation, and we define the optimization objective as follows:

$$(\theta) = \underset{(\theta)}{arg\,min} \frac{1}{2} \left\| Y \odot \left( R - \sum_{l=1}^{p} \theta^l \overset{\sim}{R}{}^{(l)} \right) \right\|_2^2 + \frac{\lambda}{2} \|\theta\|_2^2 \quad s.t.\ \theta \geq 0, \tag{5}$$

where the $\odot$ is the Hadamard product between matrices, and $Y_{u,i} = 1$ when user $u$ has rated item $i$, otherwise, $Y_{u,i} = 0$. In addition, $\| \bullet \|_p$ is the $p = 2\ L^p - norm$.

### D. PERSONALIZED RECOMMENDATION MODEL

The global recommendation model can produce the recommended results in conjunction with the users' different preferences. However, the global model assumes that the weights of all users' preferences are the same. In fact, each user has his or her own specific preferences; the global recommendation model cannot achieve personalized recommendations. For example, one person may watch a movie primarily because it features a specific actor, while someone else may be interested in the genre that the movie represents. Therefore, we extend the global model into a personalized recommendation model. Based on the different distributions of each user's preferences, we can construct different models based on personalized weights. However, it is difficult to learn the individual weights for all users because of the sparse scoring data problem: when the data are too sparse, we cannot accurately learn the weight of the users' preferences. To solve this problem, we also use similar users to help learn the weights of the target user. We calculate user similarity based on the interactions between a user and an item, which show that similar users have similar preferences. Therefore, we also have reason to believe that users with similar preferences have similar preference weights. Similarly, we modify the rating score prediction formula for the target user described above to predict the preference weights of the target user, as follows:

$$\overset{\sim}{\theta}{}^{(l)}_{u} = \frac{\sum_{v\in U, v\neq u} S^{(l)}_{u,v} \cdot \theta^{(l)}_{v}}{\sum_{v\in U, v\neq u} S^{(l)}_{u,v}} \tag{6}$$

where $S^{(l)}_{u,v}$ denotes the similarity measure between user u and user v under the path $P_l$, $\theta^{(l)}_v$ is the weight for user $v$ under the path $P_l$, and $\overset{\sim}{\theta}{}^{(l)}_{u}$ is the predicted weight for user $u$ under path $P_l$ by similar users.

We can put the above as a priori knowledge, written in the form of regularization of the objective function. Minimize the error of the weight $\theta^{(l)}$ and the predicted weight $S^{(l)} \cdot \theta^{(l)}$, as shown in the following formula:

$$\sum_{l}^{|P|} \|\theta^{(l)} - \overline{S}^{(l)} \cdot \theta^{(l)}\|_2^2 \tag{7}$$

Where $\overline{S}^{(l)}$ represents the normalization and

$$\overline{S}^{(l)}_{u,v} = \frac{S^{(l)}_{u,v}}{\sum_{v\in U, v\neq u} S^{(l)}_{u,v}}.$$

Thus, we can complete the optimization objective based on (5) and (7) as follows:

$$L(\theta) = \underset{(\theta)}{arg\,min} \frac{1}{2} \left\| Y \odot \left( R - \sum_{l=1}^{|p|} diag\left(\theta^l\right) \overset{\sim}{R}{}^{(l)}_{u} \right) \right\|_2^2 + \frac{\lambda_1}{2} \sum_{l=1}^{|P|} \left\| \theta^{(l)} - \overline{S}^{(l)} \cdot \theta^{(l)} \right\|_2^2 + \frac{\lambda_0}{2} \|\theta\|_2^2 \quad s.t.\ \theta \geq 0, \tag{8}$$

Because this objective constitutes a non-negative bound-constrained optimization problem, we can use the projected gradient method for non-negative bound-constrained optimization to solve it.

The gradient of (8) with respect to $\theta_u^{(l)}$ can be calculated as follows:

$$\frac{\partial L(\theta)}{\partial \theta_u^{(l)}} = -(Y_u \odot (R_u - \sum_{l=1}^{|p|} \theta_u^{(l)} \overset{\sim}{R}_u^{(l)})) \overset{\sim}{R}_u^{(l)T} + \lambda_1(\theta_u^{(l)} - \overset{(l)}{S}_u \theta^{(l)}) - \lambda_1 \overset{(l)T}{S}_u (\theta^{(l)} - \overset{(l)}{S} \theta^{(l)}) + \lambda_0 \theta_u^{(l)} \quad (9)$$

Algorithm 1 shows the specific process of the personalized recommendation model.

**Algorithm 1** Personalized Recommendation Model

**Input:** feedback rating matrix , heterogeneous information network $G$ (consists of several relation matrices, and we will do detailed introduction in experimental section) and regularization parameters $\lambda_0$, $\lambda_1$, update step size parameter $\alpha$ and convergence threshold $\varepsilon$ (we choose the parameters that make the result optimal by the cross validation method).

**Output:** recommendation model weighted parameters for all users and all paths

// Calculate the predicted rating based on user preferences

**for** $q \rightarrow 1toL$ **do**

Evaluate user similarity $S^{(l)}$

Calculate predict rating $R_u^{(l)}$

**end for**

// Learn Personalized Recommendation Model

Initialize $\theta > 0$

**repeat**

$\theta' = \theta$

Calculate $\frac{\partial L(\theta)}{\partial \theta}$ (Equation 9)

$\theta = \max(0, \theta - \alpha \frac{\partial L(\theta)}{\partial \theta})$

**until** $|\theta - \theta'| < \varepsilon$

## IV. EXPERIMENTS

In this section, we demonstrate the experimental performance of our algorithm by performing a set of experiments on two datasets and comparing the results with those of several other algorithms on the same datasets. Then, we perform a detailed analysis of our experimental results from three aspects.

### A. DATASETS

To illustrate the performance of the proposed algorithm, we chose two real-world datasets to perform the experimental research. The first dataset was the Douban dataset; Douban is one of the most famous community websites in China. This dataset includes 13,367 users, 12,677 movies, and user-rated movie ratings ranging from 1 to 5. The dataset also includes movie-related entities (directors, actors, genres, etc.) and relations between movies and movie-related entities. Another dataset is the Yelp Challenge dataset. This dataset includes 16,239 users and 14,284 businesses as well as ratings ranging from 1 to 5, reviews, and business information such as categories and city locations. We present a detailed description of these two datasets in Table 1.

**TABLE 1. Douban and Yelp datasets.**

| DataSet | Users | Items | Ratings | Entities | Relations |
|---|---|---|---|---|---|
| Douban | 13,367 | 12,677 | 1,068,178 | 8,798 | 72,531 |
| Yelp | 16,239 | 14,284 | 198,397 | 558 | 54,256 |

In the proposed algorithm, we need to predict the rating scores of users through the meta-paths. For the Douban Dataset, the problem involves recommending movies to users. We need to recommend movies to target users through their preferences for movie-related entities such as directors, actors, and genres. Therefore, the meta-path we choose should contain user-movie-(related entities) information. For the Yelp dataset, similarly, our problem is to recommend businesses to users. We need to consider the preferences of users for business locations, categories, and other information. Therefore, we select the meta-paths containing user-business-(related information entity) information. For these two datasets, we detail the selected meta-paths in Table 2. We determined the meta-path to be used in the dataset, which is equivalent to determining the set of node types and the set of link types in the heterogeneous information network. For example, we choose four meta-paths for the Douban dataset. Based on the four meta-paths, we can determined that the node types in the heterogeneous information network conclude user, movie, director, actor and high score movie, and the link types are user-movie, movie-director, movie-actor, user-high score. For the choice of thresholds, we can select the appropriate values according to the distribution of the data. For the two data sets selected in this experiment, we set the threshold to 30%, that is, we select the 30% highest score movies as the high scores.

**TABLE 2. Meta-path examples.**

| Dataset | Meta-Path |
|---|---|
| Douban | user-movie-user |
| | user-movie-director-movie-user |
| | user-movie-actor-movie-user |
| | user-movie (high score)-user |
| Yelp | user-company-user |
| | user-company-city-company-user |
| | user-company-category-company-user |
| | user-company (high score)-user |

### B. COMPARISON METHODS AND METRICS

To show the performance of our proposed algorithm, we compare the experimental results with those of other state-of-the-art approaches. The other three tested methods are as follows:

**TABLE 3.** Performance comparison.

| Douban | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| training set ratio / Method | 20% | | 40% | | 60% | | 80% | |
| | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE |
| PMF | 0.8081 | 0.6389 | 0.7818 | 0.6185 | 0.7765 | 0.6148 | 0.7635 | 0.6030 |
| BMF | 0.8006 | 0.6321 | 0.7679 | 0.6067 | 0.7610 | 0.6017 | 0.7578 | 0.5996 |
| SemRec | 0.7844 | 0.6054 | 0.7452 | 0.5808 | 0.7296 | 0.5689 | 0.7216 | 0.5639 |
| SPR-g | 0.7857 | 0.6017 | 0.7660 | 0.5973 | 0.7622 | 0.5967 | 0.7541 | 0.5948 |
| SPR-p | 0.7488 | 0.5736 | 0.7262 | 0.5665 | 0.7186 | 0.5483 | 0.7123 | 0.5446 |
| Yelp | | | | | | | | |
| training set ratio / Method | 20% | | 40% | | 60% | | 80% | |
| | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE |
| PMF | 1.4643 | 1.1014 | 1.4417 | 1.0817 | 1.4135 | 1.0590 | 1.3882 | 1.0368 |
| BMF | 1.3575 | 1.0060 | 1.3288 | 0.9723 | 1.3026 | 0.9513 | 1.2785 | 0.9304 |
| SemRec | 1.2358 | 0.9001 | 1.1394 | 0.8360 | 1.0853 | 0.7806 | 1.0555 | 0.7602 |
| SPR-g | 1.3247 | 0.8148 | 1.2307 | 0.8958 | 1.1921 | 0.8774 | 1.1772 | 0.8618 |
| SPR-p | 1.0299 | 0.7437 | 0.9757 | 0.7164 | 0.9508 | 0.6531 | 0.9037 | 0.6501 |

- PMF: A matrix factorization method that uses only a user-item matrix for recommendations [7].
- BMF: A matrix factorization method that uses Markov chain Monte Carlo (MCMC) methods to make approximate inferences in the model.
- SemRec: A recommendation method in the weighted HIN, which measures the similarity between users by splitting the weighted meta-path [8].

We use two widely used metrics to make the comparisons:

Root Mean Square Error (RMSE), represents the sample standard derivation of the differences between predicted values and observed values, which is calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{(u,i)\in R} (R_{u,i} - \tilde{R}_{u,i})^2}{|R|}} \tag{10}$$

and Mean Absolute Error (MAE), as the sum of two components: Quantity Disagreement and Allocation Disagreement. Quantity Disagreement is the absolute value of the Mean Error. Allocation Disagreement is MAE minus Quantity Disagreement, which is calculated as shown below:

$$MAE = \frac{\sum_{(u,i)\in R} \left| R_{u,i} - \tilde{R}_{u,i} \right|}{|R|} \tag{11}$$

The smaller the value of these two metrics indicate that the smaller the error between the predicted value and the true value, and the better the effect of the model.

### C. EXPERIMENT RESULT ANALYSIS

Table 3 lists the performances of the five methods (our global and personalized methods and the three methods listed in the previous section). For the two datasets of Douban and Yelp, we applied different training set ratios (20%, 40%, 60% and 80%) to observe the effect that data sparseness has on the above methods. For example, a training set ratio of 20% means that we randomly selected 20% of the user-item rating score data as the training data and used the remaining 80% of the data to make predictions. We used a 10-fold cross-validation method to independently perform ten random training set selections and recorded the average results. As Table 3 shows, all the methods achieved better performances on the Douban dataset compared to their performances using the same proportion of the training set on the Yelp dataset, because Yelp is sparser than Douban.

PMF is a basic matrix factorization method that uses only a user-item matrix for the recommendation. BMF is an improved collaborative filtering recommendation method based on PMF; it uses Markov chain Monte Carlo (MCMC) methods to calculate approximate inferences in the model. We used 30D feature vectors for both models. The advantage of the Bayesian MF models is that by averaging over all the parameters settings that are compatible with the data as well as the prior, they address uncertainty more effectively than do the PMF models [7]. From Table 3, we can observe that the performance of BMF is better than the performance of PMF under all conditions. However, the performance of SPR-g and SPR-p are much better than those of BMF and PMF under all conditions. The HIN-based method SemRec also achieves a better performance than the methods based on matrix factorization. This may indicate that the explanatory features based on the meta-path under HIN are more indicative of the characteristics of the users and the items.

The SemRec method is based on a weighted HIN; it combines both relation features from the Heterogeneous information network and user-item score ratings. However, it represents the user's score rating as a path weight and further divides the weighted meta-path into atomic meta-paths based on the score value. In cases where user-item score rating data are sparse, further splitting may increase the sparseness of the data; therefore, this method may not be a combination that reflects the heterogeneity of the information network and score ratings well. Its performances are worse than the performances of the proposed SPR-p on all conditions. For example, using the Douban 20% training set, SPR-p outperforms SemRec by as much as 4.75% with respect to RMSE, but on the Douban 80% training set, SPR-p outperforms SemRec by only 3.54%. By combining the results under all conditions,

we found that in the case of sparse data, the proposed method improved more.

Our proposed global recommendation model (SPR-g) combines HINs and score rating values to achieve better performance on both datasets compared to the performances of PMF and BMF. For example, compared to PMF, SPR-g improves the RMSE by 2.85% and the MAE by 6.18% on the Douban 20% training set. This comparison result shows that by including the semantic meta-path in HIN in the model it is better able to model features than can simple matrix factorization; thus, it achieves better quality recommendations. We also observe that SPR-g achieves better performances compared with PMF and BMF on sparser datasets (20% Douban, 2.85% vs 80% Douban, 1.24%), which demonstrates that the HIN–based recommendation method achieves better performances on sparse datasets.

SPR-p, the personalized recommendation method, further improves the recommendation performance under all conditions. The SPR-p method makes regularization constraints to the preference distribution of the target user based on similar users, and it provides personalized parameters for all users in the model. This approach considers the individual behavior of each user, while SPR-g treats all users the same. Compared with the global recommendation model (SPR-g), the personalized recommendation model (SPR-p) yields better performances under all conditions. Compared to SPR-g, for example, it improves the RMSE by 4.9% on the Douban 20% training set and by 28.6% on the Yelp 20% training set. This reflects the fact that different users have different personalized preference distributions. However, the global model cannot reflect these types of personalized features. The personalized model uses similar users to learn the personalized preference distribution and, thus, achieves better results. Compared with SemRec, which is the most similar to the proposed method, SPR-p also achieved better results under all conditions. On the 20% training set of both datasets, SPR-p improved the RMSE by up to 4.75% on Douban and 19.99% on Yelp. The proposed method achieved a greater improvement on the Yelp dataset, which indicates that our method has a better performance advantage on sparse datasets. This also verifies that using the score ratings as a threshold is perhaps a better option for sparse datasets than is splitting the meta-path.

Overall, the proposed method achieved better performances than the compared methods in all conditions. The results verify that combining the method with multiple preferences in recommender systems can improve recommendation quality.

### D. CASE STUDY: SPARSENESS OF USERS AND ITEM

Then, we analyze the performances of PMF, BMF, SemRec, and SPR-p under different recommendation scenarios. We performed the following experiments using the Douban 60% training set.

First, we studied the impact of user activity on model performance. We used the number of interactions between users and items to represent the degree of user activity. We divided the dataset into three groups according to the number of users' scores. Group 1 included the user with only a few scores (the average number which each user rated scores was 2); Group 3 included users with large numbers of scoring interactions (the average number which each user rated scores was 219). The results of this experiment are shown in Figure 3(a), which shows that SPR-p achieved the best performance in each group of the dataset. When the data are quite sparse (group 1), all the methods obtain poor results. However, as a number of data increases, the performance gap between each method becomes apparent. The proposed method and SemRec showed great improvements as the data amount of data increased compared to PMF and BMP. This result indicates that although the collaborative filtering method is affected by sparse datasets, adding HIN as external knowledge can alleviate the problem.

Next, we studied the impact of item prevalence on model performance. We used the number of rated items as a measure of item prevalence. Here, we also divided the dataset into three groups: Group 1 contained items with only a small number of user ratings (the average was 3), and Group 3 represents items with large numbers of user ratings (the average was 296). The result of this experiment is shown in Figure 3(b). Similar to the previous experimental results, SPR-p achieved the best performance on each dataset, but we found that, in the case of high sparsity (Group 1), SPR-p and SemRec both achieved the highest performances. Compared to the MF method, these two methods are based on the user to model characteristics; for Group 1, the dataset is sparse for items, but not for users. In reality, popular items are often liked by all kinds of people; consequently, even people with different interests are likely to be interested in the same popular items. As shown in Figure 3(c), in the Douban dataset, we chose the above four meta-paths as the user preference characteristics. For the three user groups, the proportion of the meta-path UM(R)MU is greater for the more popular items. This also shows that the user preferences for popular items cannot be judged only by their preferences for the relevant item attributes.

## V. RELATED WORKS

### A. RECOMMENDER SYSTEMS

Recommender systems are a popular research topic both in academia and industry. To better understand user behaviors, researchers have proposed many works to understand people's preferences from different perspectives [9]. Initially, these studies concentrated on explicit feedback-such as ratings; however, they now try to glean knowledge from both explicit and from implicit feedback [4], [10]. The studies on collaborative filtering solutions have also changed the research focus because a user-rating matrix is insufficient. There are also some works that have attempted to discover useful information in unstructured data (e.g., from the text of reviews or from pictures) [11], [12]. In recent studies, we find that the content-based solutions need to analyze

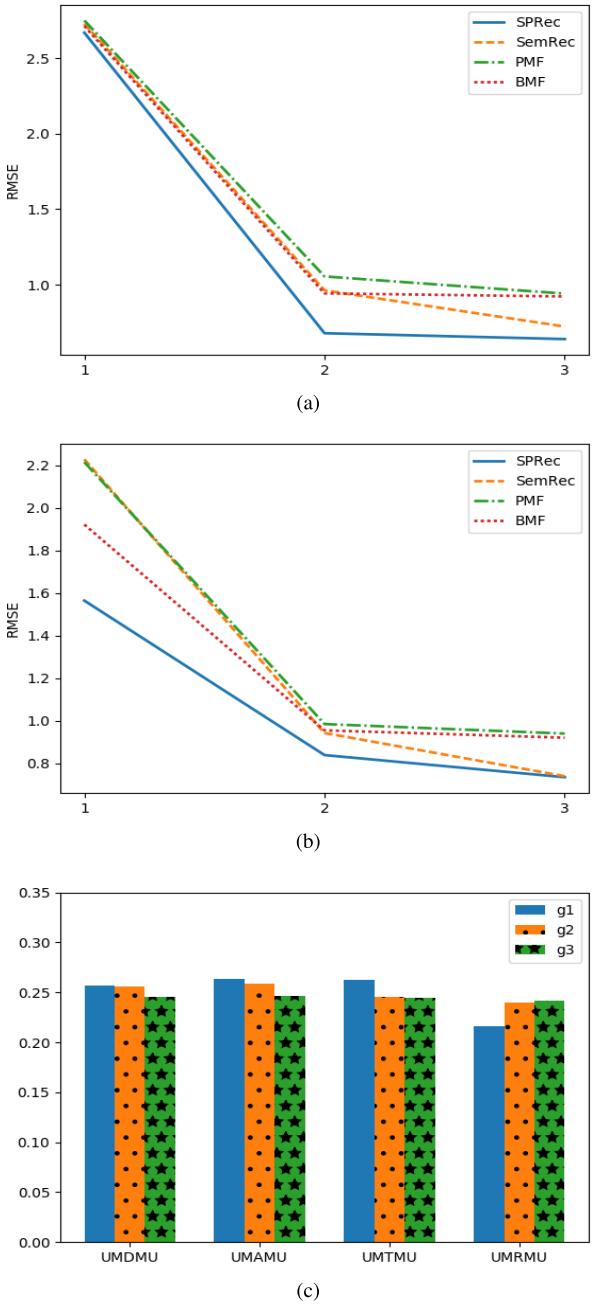


**FIGURE 3.** Performance analysis considering the sparseness of users and items: (a) performance changes with amount of user feedback, (b) performance changes with user feedback popularity, and (c) performance changes with high score preferences.

more relational data such as friend relationships or trust relationships [13], [14]. To consider additional relationships, some researchers have extracted additional information from other homogeneous networks to improve performance. Our work proposes a method that utilizes multiple relationships in heterogeneous information networks.

### B. HETEROGENEOUS INFORMATION NETWORK ANALYSIS

Recently, a new perspective on analyzing information networks was proposed by [15]. In the physical world, many types of relations and entities exist. In the past few years, information network analysis has concentrated on homogeneous data. However, many researchers believe that this is not the only way to represent information. For example, many mining tasks need the semantics that can be provided by different relations and entities; therefore, maintaining the structure of the original information is required. For example, in this study, we need to know both the social relationship and the user profiles at the same time to construct a better recommender system [16]. After heterogeneous information networks were proposed, many mining tasks were accomplished, such as the top-k search [17], similarity measurement and node embedding [18], [19]. Increasingly, studies are providing solutions to perform existing tasks better.

## VI. CONCLUSION

In this paper, we studied personalized recommendations based on user preference points in a heterogeneous information network. We implemented a similarity measure under different relations and, on this basis, defined the recommendation model at both global and individualized levels. A personalized model based on user' potential preferences was used to solve the problem of the sparse data available for individual users; this method demonstrates a more accurate use of the users' preference points. Finally, we compared the proposed method with several other algorithms and analyzed its performance under different scenarios. The proposed method achieved better results. In future work, we will test other features to help represent user preferences, such as crossing multiple heterogeneous information networks and extracting user and item characteristics from unstructured data (user reviews, item descriptions, and so on.)

## REFERENCES

[1] J. L. Herlocker, J. A. Konstan, and J. Riedl, "Explaining collaborative filtering recommendations," in *Proc. ACM Conf. Comput. Support (CSCW)*, 2000, pp. 241–250.
[2] L. Xiaozhong, "Cross social media recommendation," in *Proc. Int. AAAI Conf. Web Social Media*, 2016, pp. 1–10.
[3] X. Wang, W. Lu, and M. Ester, "Social recommendation with strong and weak ties," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, 2016, pp. 5–14.
[4] X. Yu *et al.*, "Personalized entity recommendation: A heterogeneous information network approach," in *Proc. 7th ACM Int. Conf. Web Search Data Mining (WSDM)*, 2014, pp. 283–292.
[5] S. Yizhou and Y. Sun, "Task-Guided and Path-Augmented Heterogeneous Network Embedding for Author Identification," in *Proc. 10th ACM Int. Conf. Web Search Data Mining (ACM)*, 2017, pp. 295–304.
[6] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "PathSim: Meta path-based top-K similarity search in heterogeneous information networks," in *Proc. VLDB*, 2011, vol. 3. no. 2, pp. 1–12.
[7] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo," in *Proc. ICML*, 2008, pp. 880–887.
[8] C. Shi *et al.*, "Semantic path based personalized recommendation on weighted heterogeneous information networks," in *Proc. CIKM*, 2015, pp. 453–462.
[9] T. Hofmann, "Collaborative filtering via Gaussian probabilistic latent semantic analysis," in *Proc. ACM SIGIR*, vol. 13. 2003, pp. 259–266.
[10] E. R. Núñez-Valdez, J. M. Cueva-Lovelle, O. Sanjuan-Martinez, C. E. Montenegro-Marin, and G. Infante-Hernandez, "Social voting techniques: A comparison of the methods used for explicit feedback in recommendation systems," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 1, no. 4, p. 61, 2011.

[11] C.-H. Lai and P.-R. Hong, "Group recommendation based on the analysis of group influence and review content," in *Proc. 9th Asian Conf. Intell. Inf. Database Syst. (ACIIDS)*, 2017, pp. 100–109.
[12] T. Song, Z. Peng, S. Wang, W. Fu, X. Hong, and P. S. Yu, "Review-based cross-domain recommendation through joint tensor factorization," in *Proc. 22nd Int. Conf. Database Syst. Adv. Appl. (DASFAA)*, 2017, pp. 525–540.
[13] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *Proc. 32nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2009, vol. 29A. no. 6, pp. 203–210.
[14] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 931–940.
[15] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. S. Yu. (Nov. 2015). "A survey of heterogeneous information network analysis." [Online]. Available: https://arxiv.org/abs/1511.04854
[16] F. Vahedian, "Weighted hybrid recommendation for heterogeneous networks," in *Proc. 8th ACM Conf. Recommender Syst. (RecSys)*, 2014, pp. 429–432.
[17] Y. Xiong, Y. Zhu, and P. S. Yu, "Top-K similarity join in heterogeneous information networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 6, pp. 1710–1723, Jun. 2015.
[18] C. Shi, X. Kong, Y. Huang, P. S. Yu, and B. Wu, "HeteSim: A general framework for relevance measure in heterogeneous networks," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 10, pp. 2479–2492, Oct. 2014.
[19] T. Chen, L.-A. Tang, Y. Sun, Z. Chen, and K. Zhang, "Entity embedding-based anomaly detection for heterogeneous categorical events," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2016, pp. 1396–1403. [Online]. Available: https://dl.acm.org/citation.cfm?id=3060815

**LIANG HU** received the M.S. and Ph.D. degrees in computer science from Jilin University in 1993 and 1999, respectively. He is currently a Professor and a Ph.D. supervisor with the College of Computer Science and Technology, Jilin University, China. His research areas are network security and distributed computing, including the theories, models, and algorithms of PKI/IBE, IDS/IPS, and grid computing. He is a member of the China Computer Federation.

**YU WANG** received the B.E degree in information security from Jilin University in 2015, where he is currently pursuing the M.S. degree. His research interests include data mining and machine learning.

**Zhenzhen Xie** is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Jilin University, China. Her current research area is information network analysis.

**FENG WANG** received the M.S. and Ph.D. degrees in computer science from Jilin University in 2012 and 2016. He currently holds a post-doctoral position with Jilin University. His research interests include computer networks, information security, Internet of Things, and cyber-physical Systems.