World Wide Web (2017) 20:417–435
DOI 10.1007/s11280-016-0397-x

Discerning individual interests and shared interests for social user profiling

Enhong Chen[1] · Guangxiang Zeng[1] · Ping Luo[2] · Hengshu Zhu[3] · Jilei Tian[4] · Hui Xiong[5]

Received: 30 March 2015 / Revised: 23 February 2016 /
Accepted: 27 May 2016 / Published online: 14 June 2016

Abstract Traditionally, research about social user profiling assumes that users share some similar interests with their followees. However, it lacks the studies on what topic and to what extent their interests are similar. Our study in online sharing sites reveals that besides *shared interests* between followers and followees, users do maintain some *individual interests* which differ from their followees. Thus, for better social user profiling we need to discern *individual interests* (capturing the uniqueness of users) and *shared interests* (capturing the commonality of neighboring users) of the users in the connected world. To achieve this, we extend the matrix factorization model by incorporating both individual and shared

✉ Enhong Chen
cheneh@ustc.edu.cn

✉ Guangxiang Zeng
zgx@mail.ustc.edu.cn

Ping Luo
luop@ict.ac.cn

Hengshu Zhu
zhuhengshu@baidu.com

Jilei Tian
jilei.tian@bmwna.com

Hui Xiong
hxiong@rutgers.edu

1 School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230026, China

2 Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100080, China

3 Baidu Research-Big Data Laboratory, Beijing 100085, China

4 BMW Technology, 100 North Riverside Plaza, Suite #1900, Chicago, IL 60606, USA

5 Management Science and Information Systems Department, Rutgers Business School, Rutgers University, Newark, NJ 07102, USA

interests, and also learn the multi-faceted similarities unsupervisedly. The proposed method can be applied to many applications, such as *rating prediction*, *item level social influence maximization* and so on. Experimental results on real-world datasets show that our work can be applied to improve the performance of social rating. Also, it can reveal some interesting findings, such as who likes the "controversial" items most, and who is the most influential in attracting their followers to rate an item.

Keywords Social recommendation · User profiling · Collaborative filtering · Information filtering · Social and behavioral sciences

1 Introduction

With the increasing prevalence of online social network services (SNS) and smart mobile devices, people spend more and more time on social media. The rapid growing amount of information available on social media makes it necessary to help users to select the relevant part of information that is interesting to them, thus *social recommender systems* have emerged to address this (e.g., [11, 12, 15, 16, 22]). Indeed, how to profile social users' interests plays a central role in social recommender systems.

The problem of social user profiling has attracted lots of research attentions in the past few years. A basic assumption in these works is that users' interests are similar to their followees, which is due to the effect of social influence [7]. For example, in [16], authors propose a method that factorizes the rating matrix and the trust matrix simultaneously, and the user interests matrix U are shared in both factorization. Ma et al. [15] combine the basic matrix factorization approach [18] and a social network based approach. Jamali and Ester claim that the interests propagate over the social network, thus propose a random walk model in [11] and a matrix factorization model in [12]. The random walk model proposed in [11] does not infer the users' interests directly, it predicts users' ratings of items by trust propagation and item-based method. The matrix factorization model proposed in [12] treats user's interests as a trust weighted sum of the followees' interests, and then factorizes the rating matrix.

Although the above research efforts can well explore the *shared interests* between users and their followees, most of them lose the sight of users' *individual interests* which cannot be represented by interests of their followees. For example, two users have follow relationship between them because they both like the same genre fiction, however, they may have different tastes about poetry. In order to show this, we take the book ratings from Douban[1] SNS platform as an example. Specifically, we use a vector $\mathbf{o}_u$ to denote the multidimensional (each item is treated as a dimension) interests difference between user u and her followees, which is embodied by the difference of ratings of u and the most similar ratings of her followees. Figure 1 shows the distribution of dimensional-normalized norm of $\mathbf{o}_u$ over users. From the figure, we can observe that most users have non-zero minimum difference compared with their followees. It indicates that these users have individual interests that are different from the interests of their followees. With this observation we can assume that social user's *overall interests* can be represented by two components. The first, called *shared interests*, is the interests which are similar to his followees. The other, called *individual interests*, contains the unique ingredient which is different from her followees.

To this end, we propose a novel method (**DisSUP**) for addressing the user interests profiling problem in the context of social network, which extends the matrix factorization model by incorporating both *individual interests* and *shared interests*. It is also worth mentioning

[1] http://www.douban.com/

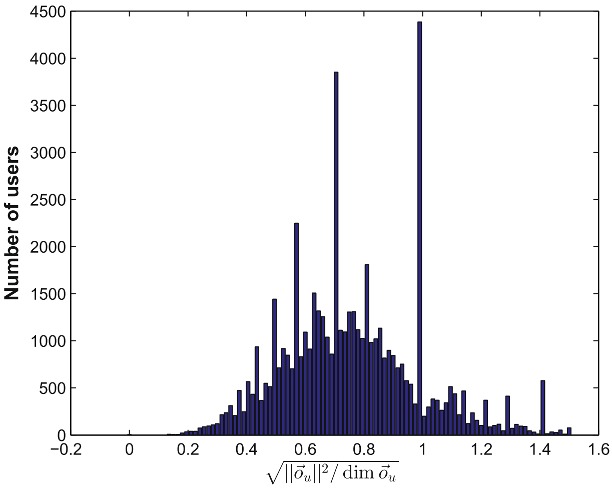


Figure 1 $o_u(i) = min_{u' \in F(u)} |r_{u,i} - r_{u',i}|$, where $\mathcal{F}(u)$ is the set of u's followees, $r_{u,v}$ is the rating of item v posted by u

that most of the previous works (e.g., [11, 12, 15, 16]) assume single and homogeneous relationships between users. However, the relationships between users are multi-faceted and heterogeneous. Users may follow different users on social networks for different reasons, e.g., they are friends, they share interests in sports, they like the same movies. In a recent study for social rating in [22], categories of items are regarded as the facets of items and are used to embody the multi-faceted relationships between users. However, there are two drawbacks: 1) the categories are defined manually which is usually not only human-labor costly but also unavailable in most situations; 2) it is inevitable that the pre-defined categories are not independent to each other, and the method proposed in [22] leaves out the correlations between the pre-defined categories. To address these two issues, in our approach we also introduce unsupervised multi-faceted social relationships, and learn them automatically. Now we summarize the contributions of this paper as follows:

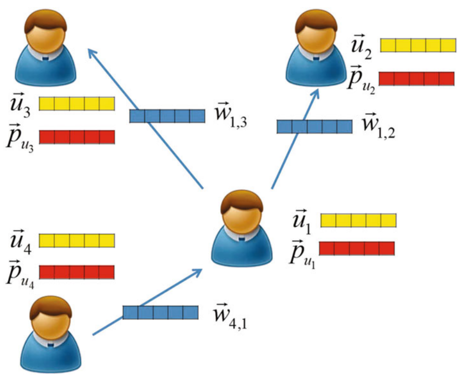
- We propose a novel method (**DisSUP**) for discerning *individual interests* and *shared interests* in the context of social network.
- We propose three application scenarios for applying our proposed method.
- We crawl two large-scale datasets from the Douban SNS Web site for evaluating the effectiveness of the proposed model, and the experimental results on the three proposed applications show that the proposed method outperforms the baseline methods.

The rest of the paper is organized as follows. We propose our problem and introduce necessary preliminaries in Section 2. Then in Section 3, we formulate the **DisSUP** model, describe how to solve this model, and provide a complexity analysis for the proposed methods. In Section 4, we discuss 3 applications using our proposed model. Section 5 presents experimental results and findings. Section 6 reviews related work. Finally, in Section 7, we conclude this paper.

2 Problem formulation and preliminaries

In this section we formally define our research problem, and then introduce some preliminaries of our approach. First of all, let's take a look at the main idea of our method.

Figure 2 A snap shot of social network with unsupervised multi-faceted social correlations and individual interests



We regard the user interests expressed in the ratings as *overall interests*, and the interests owned by the users uniquely as *individual interests*. In Figure 2, the *overall interests* are represented by a vector $\mathbf{u}$ and the *individual interests* by vector $\mathbf{p}_u$. Here, we treat each dimension of $\mathbf{u}$ as a "facet" of users' interests. When a user u_i follows another user u_j in a social network, their correlations on interests are represented by a vector $\mathbf{w}_{i,j}$. The elements of $\mathbf{w}_{i,j}$ are all in range of (0, 1) and they measure the similarities between user u_i and u_j in the corresponding facets. Finally, all these vectors will be learned by solving the model proposed in Section 3.

Problem formulation Given an incomplete rating matrix R, and a social network $G = (\mathcal{U}, \mathcal{E})$. We aim at inferring the *overall interests* of users and their *individual interests* which are different from the interests of their followees.

Before we get into further analysis, we introduce some preliminaries here. Most mathematical Notations used in this paper are summarized in Table 1, besides: 1) $\mathbf{n} = \mathbf{e}^2$ means that $n_i = e_i^2$; 2) let $S(\cdot)$ be a function with a scalar input, and $\mathbf{w} = S(\mathbf{e})$ means that $w_i = S(e_i)$.

Table 1 Mathematical Notations

$\mathcal{U}$	The set of users.
$\mathcal{V}$	The set of items.
$u \in \mathcal{U}$	A user.
$v \in \mathcal{V}$	An item.
R	Incomplete rating matrix.
Ω	The set of known value positions of R.
$\mathcal{E}$	The set of edges.
G	A social network, $G = (\mathcal{U}, \mathcal{E})$.
$e_{u,u'} \in \mathcal{E}$	An edge which represents that user u' is followed by user u.
$\mathcal{F}(u)$	The set of users which is followed by u.
$\mathcal{D}(u)$	The set of users who follow u.
$\odot$	$\mathbf{n} = \mathbf{e} \odot \mathbf{u}$, then $n_i = e_i \cdot u_i$.
$\oslash$	$\mathbf{n} = \mathbf{e} \oslash \mathbf{u}$, then $n_i = e_i \div u_i$.
$\partial_\bullet$	$\mathbf{n} = \frac{\partial_\bullet \mathbf{h}}{\partial_\bullet \mathbf{e}}$, then $n_i = \frac{\partial h_i}{\partial e_i}$.

Latent factor model Latent factor models are popular with collaborative filtering rating prediction problems where the goal is to uncover latent features that explain observed ratings, examples include pLSA [10], neural networks [19] and latent dirichlet allocation [1]. We will focus on models which are introduced by Singular Value Decomposition (SVD) on the user-item rating matrix, such as Probabilistic Matrix Factorization (PMF) [18]:

$$\min \sum_{(u,v)\in\Omega} (r_{uv} - \mathbf{u}\cdot\mathbf{v})^2 + \lambda\Big(\sum_{u\in\mathcal{U}} ||\mathbf{u}||^2 + \sum_{v\in\mathcal{V}} ||\mathbf{v}||^2\Big). \tag{1}$$

In the above model, we can regard the latent features $\mathbf{u}$ as the interests of user u. While in the context of social network, users follow people who they are interested in, so the interests of users are not identical to each other. In order to formulate the interests correlation between users, we introduce the concept of *interest distance*.

Interest distance Ideally, if a user is fully correlated with his followees, his *overall interests* $\mathbf{u}$ can be decomposed into two parts, namely his *individual interests* $\mathbf{p}_u$ and the *shared interests* $\mathbf{u} - \mathbf{p}_u$ which are in common with the interests of his followees. Here we introduce the concept *interest distance* which measures the correlation between users and their followees, this measure should be small (ideally 0) if the user is fully correlated with his followees. Let $\mathbf{w}_{u,u'}$ measure what and how much interest user u has in common with user u' whom he follows, whose elements are all in the range (0, 1). The more u and u' are similar to each other in facet k, the closer the k-th element of $\mathbf{w}_{u,u'}$ approaches to "1". Otherwise, the more u is different from u' in facet k, the closer the k-th element of $\mathbf{w}_{u,u'}$ approaches to "0". The shared interests between the two users can be represented as $\mathbf{q}_{u,u'} = \mathbf{w}_{u,u'} \odot \mathbf{u}'$. We define the *interest distance* of u and his followees as follows.

Definition 1 The *interest distance* of u and his followees is defined as:

$$Dist(u) = \sum_{u'\in\mathcal{F}(u)} \mathbf{h}_{u,u'} \cdot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2, \tag{2}$$

where $\mathbf{h}_{u,u'} = \mathbf{w}_{u,u'} \oslash \sum_{u''\in\mathcal{F}(u)} \mathbf{w}_{u,u''}$, $\mathbf{w}_{u,u'} = S(\mathbf{e}_{u,u'})$, and the scale function $S(x) = \frac{1}{1+e^{-x}}$.

In addition, if a user is fully correlated with his followees, interest distance between him and his followees should be small (ideally 0). We claim that $\mathbf{w}_{u,u'}$ measures the similarity of interests between users u and u'. The following is important as we have mentioned in the previous paragraph: 1) the more u and u' are similar to each other in facet k, the closer the k-th element of $\mathbf{w}_{u,u'}$ approaches to "1"; 2) otherwise, the more u is different from u' in facet k, the closer the k-th element of $\mathbf{w}_{u,u'}$ approaches to "0".

3 Discerning user interests

We choose the latent factor model in (1), and combine the *interest distance*, then we formulate the optimization problem in (3):

$$\begin{aligned} \min f &= \tfrac{1}{2}\sum_{\Omega}(\tilde{r}_{u,v} - r_{u,v})^2 + \tfrac{\gamma}{2}\sum_{\mathcal{U}} Dist(u) \\ &+ \tfrac{\lambda}{2}\Big(\sum_{\mathcal{V}} ||\mathbf{v}||^2 + \sum_{\mathcal{U}}(||\mathbf{u}||^2 + ||\mathbf{p}_u||^2)\Big). \end{aligned} \tag{3}$$

The first summing term $\frac{1}{2}\sum_{\Omega}(\tilde{r}_{u,v} - r_{u,v})^2$ infers users' overall interests $\{\mathbf{u}\}$ and item profiles $\{\mathbf{v}\}$ by minimizing the difference between the predicted ratings $\{\tilde{r}_{u,v}\}$ and the true ratings $\{r_{u,v}\}$. The second summing term $\frac{\gamma}{2}\sum_{\mathcal{U}} Dist(u)$ extracts individual interests and users' correlations by minimizing the interests distance between users and their followees. The third one is the regularized term which prevents the model from overfitting.

As we can see that the problem is non-convex and we cannot find the solution analytically, so we propose to solve the problem numerically by gradient descent method [2].

3.1 Gradient computation

Let $\Omega(u)$ denote the set of items that are rated by u, and $\Omega(v)$ denote the set of users that is rated v. The gradient on $\mathbf{v}$ for any $v \in \mathcal{V}$ can be computed as:

$$\frac{\partial f}{\partial \mathbf{v}} = \lambda \mathbf{v} + \sum_{u \in \Omega(v)} (\tilde{r}_{u,v} - r_{u,v})\mathbf{u}. \tag{4}$$

Then the gradient on $\mathbf{p}_u$ and $\mathbf{e}_{u,u'}$ can be computed as:

$$\frac{\partial f}{\partial \mathbf{p}_u} = \lambda \mathbf{p}_u + \gamma(\mathbf{p}_u - \mathbf{u} + \sum_{u' \in \mathcal{F}(u)} \mathbf{h}_{u,u'} \odot \mathbf{q}_{u,u'}), \tag{5}$$

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{e}_{u,u'}} &= \frac{\gamma}{2}[(\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2 \odot \frac{\partial_\bullet \mathbf{h}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}} \\ &\quad -2\mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}) \odot \mathbf{u}' \odot \frac{\partial_\bullet \mathbf{w}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}} \\ &\quad + \sum_{u'' \in \mathcal{F}(u), u'' \neq u'} (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u''})^2 \odot \frac{\partial_\bullet \mathbf{h}_{u,u''}}{\partial_\bullet \mathbf{e}_{u,u'}}], \end{aligned} \tag{6}$$

$$\frac{\partial_\bullet \mathbf{h}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}} = [(1 - \mathbf{h}_{u,u'}) \oslash \sum_{u'' \in \mathcal{F}(u)} \mathbf{w}_{u,u''}] \odot \frac{\partial_\bullet \mathbf{w}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}}, \tag{7}$$

$$\frac{\partial_\bullet \mathbf{h}_{u,u''}}{\partial_\bullet \mathbf{e}_{u,u'}} = [-\mathbf{h}_{u,u''} \oslash \sum_{u''' \in \mathcal{F}(u)} \mathbf{w}_{u,u'''}] \odot \frac{\partial_\bullet \mathbf{w}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}}, u'' \neq u'. \tag{8}$$

Then we substitute (7) and (8) into (6) we obtain

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{e}_{u,u'}} &= \frac{\gamma}{2} \frac{\partial_\bullet \mathbf{w}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}} \odot \{[(\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2 \\ &\quad - \sum_{u'' \in \mathcal{F}(u)} \mathbf{h}_{u,u''} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u''})^2] \oslash \sum_{u'' \in \mathcal{F}(u)} \mathbf{w}_{u,u''} \\ &\quad -2\mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}) \odot \mathbf{u}'\} \end{aligned} \tag{9}$$

$$\frac{\partial_\bullet \mathbf{w}_{u,u'}}{\partial_\bullet \mathbf{e}_{u,u'}} = \frac{\partial S(\mathbf{e}_{u,u'})}{\partial \mathbf{e}_{u,u'}} = \mathbf{w}_{u,u'} \odot (1 - \mathbf{w}_{u,u'}). \tag{10}$$

Finally, the gradient on $\mathbf{u}$ is:

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{u}} &= \lambda \mathbf{u} + \sum_{v \in \Omega(u)} (r_{u,v} - \tilde{r}_{u,v})\mathbf{v} + \gamma \sum_{u' \in \mathcal{F}(u)} \mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}) \\ &\quad - \gamma \sum_{u' \in \mathcal{D}(u)} \mathbf{h}_{u',u} \odot (\mathbf{u}' - \mathbf{p}_{u'} - \mathbf{q}_{u',u}) \odot \mathbf{w}_{u',u}. \end{aligned} \tag{11}$$

To this end, let $U = \{\mathbf{u}\}$, $P = \{\mathbf{p}_u\}$, $V = \{\mathbf{v}\}$ and $E = \{\mathbf{e}_{u,u'}\}$, then we can compute the gradient $\frac{\partial f}{\partial [U,P,V,E]}$ through (11), (5), (4) and (9).

3.2 Computation complexity analysis

Here we mainly discuss about the time complexity of computing the gradient of the function in (3) in each iteration. Assuming that the number of latent features is K. We notice that the function in (3) is the sum of terms $\{\frac{1}{2}(\tilde{r}_{u,v} - r_{u,v})^2\}$, $\{\frac{\gamma}{2}Dist(u)\}$, $\{\frac{\lambda}{2}||\mathbf{v}||^2\}$, $\{\frac{\lambda}{2}||\mathbf{u}||^2\}$ and $\{\frac{\lambda}{2}||\mathbf{p}_u||^2\}$. Each term only involves a few variables, for example, $\frac{1}{2}(\tilde{r}_{u,v} - r_{u,v})^2$ involves $\mathbf{u}$ and $\mathbf{v}$, $\frac{\gamma}{2}Dist(u)$ involves $\mathbf{u}$, $\mathbf{p}_u$ and $\{\mathbf{u}', \mathbf{e}_{u,u'}|u' \in \mathcal{F}(u)\}$. For each term, non-zero sub gradient only exists on the involved variables, so we just compute that part of the gradient. As the function in (3) is the sum of the terms, so the sum of the sub gradients is the gradient of the function in (3).

The above analysis indicates that we can compute the gradient by the following way: 1) set the gradient variables to 0 initially; 2) add the non-zero sub gradients of the terms mentioned above to the corresponding gradient variables sequentially (we call "update the gradient"); 3) when all the terms pass through, we get the true gradient. Now we explore the computation of the gradient.

Assuming that we have already computed the value of function in (3) and all the middle variables $\{\tilde{r}_{u,v}\}$, $\{\mathbf{w}_{u,u'}\}$, $\{\mathbf{h}_{u,u'}\}$, $\{\mathbf{q}_{u,u'}\}$, and $\{\sum_{u''\in\mathcal{F}(u)} \mathbf{w}_{u,u''}\}$ are known values. The above analysis indicates us that we can compute $\frac{\partial f}{\partial [U,P,V,E]}$ sequentially: 1) set $\frac{\partial f}{\partial [U,P,V,E]} = \mathbf{0}$; 2) update $\frac{\partial f}{\partial [U,P,V,E]}$ by the sub gradients of the terms $\{\frac{1}{2}(\tilde{r}_{u,v} - r_{u,v})^2\}$, $\{\frac{\gamma}{2}\mathcal{D}(u)\}$, $\{\frac{\lambda}{2}||\mathbf{v}||^2\}$, $\{\frac{\lambda}{2}||\mathbf{u}||^2\}$, $\{\frac{\lambda}{2}||\mathbf{p}_u||^2\}$ and $\{\frac{\lambda}{2}||\mathbf{e}_{u,u'}||^2\}$ sequentially; 3) when all the terms pass through, we get the true gradient. Next, we analyze the update process of each term:

1. $\{\frac{\lambda}{2}||\mathbf{v}||^2\}$, $\{\frac{\lambda}{2}||\mathbf{u}||^2\}$ and $\{\frac{\lambda}{2}||\mathbf{p}_u||^2\}$:
$\frac{\lambda}{2}||\mathbf{v}||^2$: $\triangle\frac{\partial f}{\partial \mathbf{v}} = \lambda\mathbf{v}$, $\{\frac{\lambda}{2}||\mathbf{v}||^2\}$ need $|\mathcal{V}| \cdot K$ operations.
$\frac{\lambda}{2}||\mathbf{u}||^2$: $\triangle\frac{\partial f}{\partial \mathbf{u}} = \lambda\mathbf{u}$, $\{\frac{\lambda}{2}||\mathbf{u}||^2\}$ need $|\mathcal{U}| \cdot K$ operations.
$\frac{\lambda}{2}||\mathbf{p}_u||^2$: $\triangle\frac{\partial f}{\partial \mathbf{p}_u} = \lambda\mathbf{p}_u$, $\{\frac{\lambda}{2}||\mathbf{p}_u||^2\}$ need $|\mathcal{U}| \cdot K$ operations.

2. $\frac{1}{2}(\tilde{r}_{u,v} - r_{u,v})^2$:

$$\triangle\frac{\partial f}{\partial \mathbf{u}} = (\tilde{r}_{u,v} - r_{u,v})\mathbf{v}, \tag{12}$$

$$\triangle\frac{\partial f}{\partial \mathbf{v}} = (\tilde{r}_{u,v} - r_{u,v})\mathbf{u}. \tag{13}$$

Each of the above two updates needs K operations, so for all $\{\frac{1}{2}(\tilde{r}_{u,v} - r_{u,v})^2\}$ we need $2|\Omega| \cdot K$ operations.

3. $\frac{\gamma}{2}Dist(u)$:

First we break the user represented term $Dist(u)$ into edge represented terms $\{\mathbf{h}_{u,u'} \cdot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2\}$. Then for each edge term $\frac{\gamma}{2}\mathbf{h}_{u,u'} \cdot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2$, by observation in (5), (11) and (9), we update:

$$\triangle\frac{\partial f}{\partial \mathbf{p}_u} = -\gamma\mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}), \tag{14}$$

$$\triangle \frac{\partial f}{\partial \mathbf{u}} = \gamma \mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}), \tag{15}$$

$$\triangle \frac{\partial f}{\partial \mathbf{u}'} = -\gamma \mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}) \odot \mathbf{w}_{u,u'}, \tag{16}$$

$$\begin{aligned}\triangle \frac{\partial f}{\partial \mathbf{e}_{u,u'}} &= \frac{\gamma}{2} \mathbf{w}_{u,u'} \odot (1 - \mathbf{w}_{u,u'}) \odot \{[(\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'})^2 \\ &- \sum_{u'' \in \mathcal{F}(u)} \mathbf{h}_{u,u''} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u''})^2] \oslash \sum_{u'' \in \mathcal{F}(u)} \mathbf{w}_{u,u''} \\ &- 2\mathbf{h}_{u,u'} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u'}) \odot \mathbf{u}'\}.\end{aligned} \tag{17}$$

We notice that for different $u' \in \mathcal{F}(u)$, all $\triangle \frac{\partial f}{\partial \mathbf{e}_{u,u'}}$ share a same $\sum_{u'' \in \mathcal{F}(u)} \mathbf{h}_{u,u''} \odot (\mathbf{u} - \mathbf{p}_u - \mathbf{q}_{u,u''})^2$, whose computation needs $|\mathcal{F}(u)| \cdot K$ operations. Each of the above 4 updates need K operations, so for only $\frac{\gamma}{2} Dist(u)$ alone we need $5|\mathcal{F}(u)| \cdot K$ operations. Then for all $\{\frac{\gamma}{2} Dist(u)\}$ we need $5 \sum_{u \in \mathcal{U}} |\mathcal{F}(u)| \cdot K = 5|\mathcal{E}| \cdot K$ operations.

Finally, the computation complexity of the gradient $\frac{\partial f}{\partial [U,P,V,E]}$ is: $K \cdot (2|\Omega| + 5|\mathcal{E}| + 2|\mathcal{U}| + |\mathcal{V}|)$. We summarize the above gradient computation procedure in Algorithm 1. The proposed social user profiling algorithm is summarized in Algorithm 2, which is called **DisSUP** (Discerning Individual interests and Shared interests for Social User Profiling).

Algorithm 1 Compute Gradient

Input: $\Omega = \{(u, v, r_{u,v})\}$, $G = (\mathcal{U}, \mathcal{E})$, $X = [U, P, V, E]$;
Output: $\frac{\partial f}{\partial X}$;
1: Initialize $\frac{\partial f}{\partial X} = \lambda \cdot [U, P, V, 0]$;
2: For each rating record $(u, v, r_{u,v})$, update $\frac{\partial f}{\partial \mathbf{u}}$, $\frac{\partial f}{\partial \mathbf{v}}$ by Equations (12) and (13);
3: For each edge $e_{u,u'}$, update $\frac{\partial f}{\partial \mathbf{p}_u}$, $\frac{\partial f}{\partial \mathbf{u}}$, $\frac{\partial f}{\partial \mathbf{u}'}$, $\frac{\partial f}{\partial \mathbf{e}_{u,u'}}$ by Equations (14), (15), (16) and (17);
4: **return** $\frac{\partial f}{\partial X}$;

Algorithm 2 Social User Profiling (DisSUP)

Input: $\Omega = \{(u, v, r_{u,v})\}$, $G = (\mathcal{U}, \mathcal{E})$, ε, δ, $MaxIter$;
Output: $U = \{\mathbf{u}\}$, $P = \{\mathbf{p}_u\}$, $V = \{\mathbf{v}\}$ and $W = \{\mathbf{w}_{u,u'}\}$;
1: Random initialize $X = [U, P, V, E]$, $W = S(E)$, $t = 0$;
2: Compute cost f by Equation (3);
3: Compute gradient $\frac{\partial f}{\partial X}$ by Algorithm 1;
4: **while** $||\frac{\partial f}{\partial X}|| > \varepsilon$ and $t < MaxIter$ **do**
5: Update $X = X - \delta \cdot \frac{\partial f}{\partial X}$, $W = S(E)$;
6: Compute cost f by Equation (3);
7: Compute gradient $\frac{\partial f}{\partial X}$ by Algorithm 1;
8: $t = t + 1$;
9: **end while**
10: **return** U, P, V and W;

4 Applications of DisSUP

In this section, we introduce three potential application scenarios of our proposed **DisSUP** model.

4.1 Social rating prediction

The rating prediction is a natural application of our proposed model, since the model makes use of the rating data for profiling interests of social users. Once we have the model trained, we can predict the rating that user u rates on item v by the equation:

$$\tilde{r}_{u,v} = \mathbf{u} \cdot \mathbf{v}. \tag{18}$$

It is worth noting that our model can learn the overall interests for those users that without any ratings in the training data (cold start users), provided that those users have at least one followee or follower. That is because when a user has at least one followee or follower, there is at least one *interests distance term* "$Dist(u)$" that contains the overall interests of the cold start user.

4.2 Controversial item recommendation

First we define what items are regarded as controversial items. Generally, the items are called controversial if neighboring users always rate them differently. For a given item v we can find all the pairs of neighboring users who both rate this item v. Namely, this set is denoted as

$$\mathcal{NU}_v = \{(u', u) | u' \text{ follows} u \text{ and they both rate item} v\}.$$

Then, we can calculate the following value for v:

$$contro_v = \frac{\sum_{(u',u)\in\mathcal{NU}_v} |r_{u',v} - r_{u,v}|}{|\mathcal{NU}_v|}. \tag{19}$$

The bigger the value of $contro_v$, the more controversial the ratings of two neighboring users on item v. Then, for a given threshold β, an item v is controversial when $contro_v > \beta$.

Then, for these controversial items we try to guess who may like them most. As the two neighboring users rate differently on these items the individual interests of users may work better on these items. Thus, with the individual interests vector $\mathbf{p}_u$ we can use the following equation to measure the rating of u on any controversial item v:

$$\theta_{u,v} = \mathbf{p}_u \cdot \mathbf{v}. \tag{20}$$

4.3 1-Hop influential user identification

In this application, we aim at identifying the most influential user to her immediate followers. In other words, for a given item v we want to identify the users such that after they rate this item most of their followers will rate the same item similarly.

We claim that the user's influence is multi-faceted, and we should choose the people who are the most influential in the facets of the item. In our model, $\mathbf{w}_{u',u}$ measures the multi-faceted similarity between user u' and his followee u. The shared interest $\mathbf{u} \odot \mathbf{w}_{u',u}$ can be treated as the influence of u to her follower u'. Thus, we define the multi-faceted influence of user u to her followers as:

$$\rho_u = \mathbf{u} \odot \sum_{u'\in\mathcal{D}(u)} \mathbf{w}_{u',u}. \tag{21}$$

Then, for a given item v, user u's 1-hop influence on item v is defined as follows:

$$\varphi_{u,v} = \rho_u \cdot \mathbf{v}. \tag{22}$$

The higher the $\varphi_{u,v}$, the more influence user u affect her followers on item v.

Table 2 Basic statistics of datasets

Dataset	#user	#item	#rating	#follow relations
Book	80,483	387,863	4,855,906	6,222,882
Movie	83,129	81,210	14,977,942	6,456,991

5 Experimental results

Datasets Our datasets are collected from the Douban[2] Web site, which is a Chinese SNS website allowing registered users to record information and create content related to books, movies and music. Users can follow others and receive shared content from their followees. We collected the books and movies subsets from the Web site, each is started with 10 seeds users. Specifically, users are collected by breadth-first searching strategy and stop after 3 steps forward from the seeds. We maintained the follow relationship and collected all the ratings of the users of the two subsets. Each rating is attached with a time stamp. We sorted the ratings by the time stamp and split the rating data into training set and test set (80 % for training, 20 % for test) so that the time stamps of the ratings in the training set are always earlier than that of the testing set. Table 2 shows the basic statistics of the two datasets.

Baseline methods We choose two baseline methods for comparison, namely PMF [18] and SocialMF [12]. The PMF model is formulated as (1). The SocialMF model is formulated as follows:

$$\begin{aligned} \min f &= \frac{1}{2}\sum_{\Omega}(\tilde{r}_{u,v} - r_{u,v})^2 \\ &+\frac{\gamma}{2}\sum_{\mathcal{U}}||\mathbf{u} - \sum_{\mathcal{F}(u)} T_{u,u'}\mathbf{u}'||^2 \\ &+\frac{\lambda}{2}(\sum_{\mathcal{V}}||\mathbf{v}||^2 + \sum_{\mathcal{U}}||\mathbf{u}||^2), \end{aligned} \tag{23}$$

where $T_{u,u'}$ is the trust weight that measures how much trust the user u pays to his followee u'. When a user u follows another user u', u expresses positive trust to u', we set $T_{u,u'} = \frac{1}{|\mathcal{F}(u)|}$ according to [12]. These two methods can be used for the applications of *social rating prediction* and *controversial item recommendation* directly. For the *1-hop influential user identification* application, we use the following solutions for the PMF and SocialMF methods, which is similar to (21) and (22),

$$\eta_{u,v} = \mathbf{t}_u \cdot \mathbf{v}, \text{where} \mathbf{t}_u = \sum_{u' \in \mathcal{D}(u)} T_{u',u}\mathbf{u}. \tag{24}$$

For our method, we denote our method as DisSUP$_u$ when the overall interest vector used as shown in (18), and denote our method as DisSUP$_p$ when the individual interest vector used as shown in (20).

Evaluation metrics For the *social rating prediction* application, we use the RMSE (Root Mean Square Error) metric. For the *controversial item recommendation* and *1-hop influential user identification* applications, we use the NDCG [24] metric: first we define *Discounted Cumulative Gain* (DCG) for a sequence $\{rel_i\}$. Let IDCG$_n$ be the DCG of

[2] http://www.douban.com/

the ideal sequence $\{rel'_i\}$ that $rel'_i > rel'_j$ when $i < j$, and all the definitions are shown in (25).

$$DCG_n = rel_1 + \sum_{i=2}^{n} \frac{rel_i}{\log_2 i}, NDCG = \frac{DCG_n}{IDCG_n}. \quad (25)$$

For the evaluation of *controversial item recommendation* application, rel_i is the rating that the i-th user rated for the selected controversial item. For the evaluation of *1-hop influential user identification* application, we first select an item, then we guess who is the most influential user to his followers about the selected item, and rel_i is the number of the rank i-th user's followers who have ratings on the selected item in the testing set.

Experiment settings We use K = 5 and 10 for the number of hidden features, respectively. We first use PMF method for tuning the λ parameter. In our preliminary experiments, we set $\lambda \in \{10^{-5}, 10^{-4}, 10^{-3}, 0.01, 0.1, 1.0, 10.0,$
$100.0\}$, and we found that the PMF method performs the best at $\lambda = 10^{-3}$ on both hidden feature number settings for both data sets. So we fix $\lambda = 10^{-3}$ for all the methods in the experiment. In addition, we pre-processed the training data before conducting the methods and recover the predicted values before evaluation. These details can be found in the supplementary materials.

5.1 Social rating prediction

We first study the impact of γ on the results, then we compare the best overall results, finally we compare the results on cold start users.

Impact of γ on the results To show the impact of γ on the results, we let γ for SocialMF (denoted as $\gamma_{SocialMF}$) change in the range (0, 10], and the γ for DisSUP$_u$ (denoted as γ_{DisSUP_u}) change in the range (0, 100]. Figure 3 shows the RMSE on test data as the changing of the γ parameter for methods SocialMF and DisSUP$_u$. The results are similar for both latent feature settings on both datasets. For the SocialMF method, the result improves as γ increases first, then after it reaches its best performance the result becomes worse as γ increases. For our DisSUP$_u$ method, the variation of the result is much smoother than that of SocialMF, when the γ become sufficient large, the result of DisSUP$_u$ is better than the best result of SocialMF and they change slowly as γ increases. Finally, the best perform γ setting for the two methods are shown in Table 3.

Best overall results comparison Table 4 shows the best performance of the baseline methods and our method, which means the λ parameter is set to 10^{-3} and the γ parameter settings for SocialMF and DisSUP$_u$ are according to Table 3. As we can see from the table, for the overall RMSE comparison, our DisSUP$_u$ method performs the best on both datasets with different settings of K, which shows that the discerning of individual and shared interests can improve the overall interests learning for profiling social users. We also can see that DisSUP$_u$ method outperforms the PMF method, which validates that the incorporation of social network information can improve the results while making recommendation in the context of social networks.

Performance on cold start users. After careful investigation, we found that there are 6,962 users in the book subset and 6,924 users in the movie subset without any ratings in

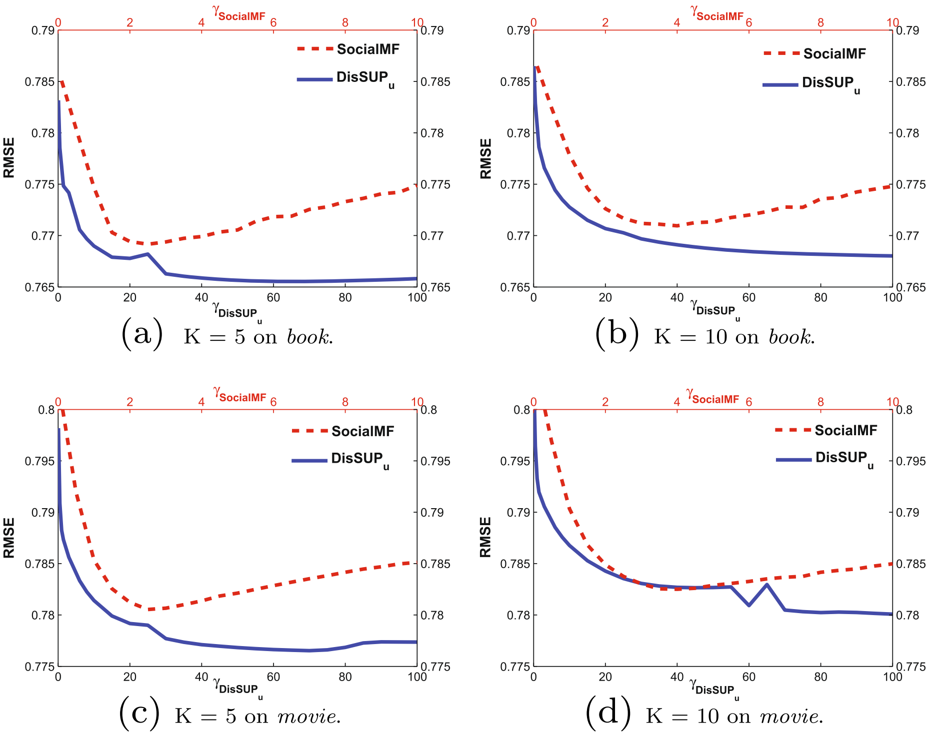


Figure 3 Impact of γ on results

the training data, but they have 160,373 and 410,271 ratings in the test data respectively. We evaluate the performance of different methods on these cold start users.

As shown in Table 5, our DisSUP$_u$ method still performs the best. Actually, the predictions of PMF method to the cold start users are all the average rating of the training data set, so there is no differences between the settings $K = 5$ and $K = 10$. Both SocialMF and DisSUP$_u$ methods outperform PMF method, which proves that the social relation help improving the recommendation for cold start users. As the performances of DisSUP$_u$ are better than that of SocialMF, the multi-faceted social relationship outperforms the single one for recommendation of cold start users.

5.2 Controversial item recommendation

For this application, we first use the training data to compute the $contro_v$ in (19) for each item v in the test set, then we set the the controversial threshold $\beta = 1.0$, and let

Table 3 The Best γ settings

	Book		Movie	
	K = 5	K = 10	K = 5	K = 10
SocialMF	2.5	4.0	2.5	4.0
DisSUP$_u$	65.0	100.0	65.0	100.0

Table 4 Best Overall RMSE comparison

Data set		PMF	SocialMF	DisSUP_u
Book	K = 5	0.786166	0.769162	0.765535
	K = 10	0.786765	0.770955	0.768023
Movie	K = 5	0.800180	0.780537	0.776529
	K = 10	0.802928	0.782508	0.780079

$|\mathcal{NU}_v| \geq 10$. Besides, we further select those controversial items which have 10 ratings in the testing data at least, because we need to rank the users' preference for the selected controversial items. Finally, we get 1,208 controversial books and 1,023 controversial movies.

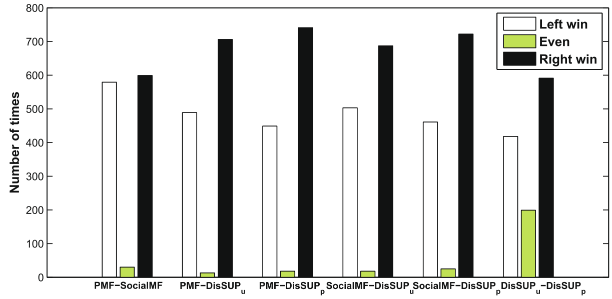
Here we denote the ranking method that utilizes the formula in (20) as DisSUP_p. The other ranking methods rank the candidate user list by their predicting rating scores for the users. Figure 4 shows the paired NDCG comparison results of *controversial item recommendation*. Generally speaking, the DisSUP_p method performs the best, DisSUP_u method outperforms both PMF and SocialMF methods and SocialMF performs better than PMF method. The only exception occurs on the *movie* dataset when $K = 10$ in the comparison "PMF-SocialMF", the PMF method outperforms the SocialMF method, we think the possible explanation for this is the single correlation between users might hurt the representation of users' interests on controversial items. In addition, the number of "Even" in the comparison of "DisSUP_u-DisSUP_p" is much larger than that of the other comparisons, that is because the individual interests are part of overall interests, so these two methods are more similar than the other pairs. The DisSUP_p method wins in every comparison with the other methods, which indicates that the individual interests are more effective in *controversial item recommendation*. Table 6 shows the average NDCG comparison of the 4 methods, our DisSUP_p method outperforms the other methods. Although the differences are small (at 10^{-3} level), they still show that the individual interests are more effective for finding users that might like the controversial items.

5.3 1-Hop influential user identification

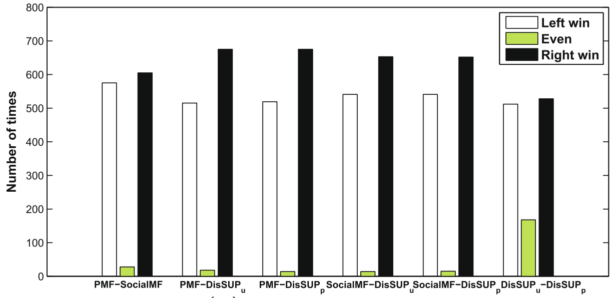
For this application, we aim at identifying the most influential user to her immediate followers. In other words, for a given item v we want to identify the users such that after they rate this item most of their followers will rate the same item. To this end, we need to select those items whose first ratings in the training data are not too old before the latest rating of the training data, and the numbers of ratings in training data are not too small. Thus, we first select those items that the time interval between their earliest ratings' time stamp and the latest time stamp of the training set is less than 12 weeks and each of the items has at

Table 5 Performance on cold start users

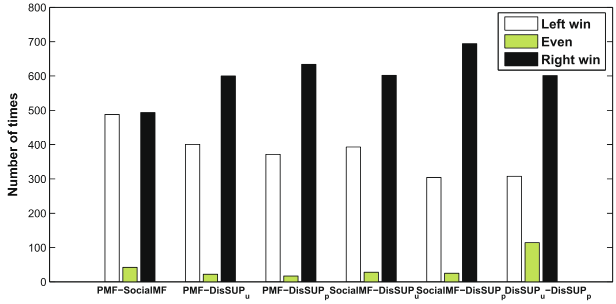
Data set		PMF	SocialMF	DisSUP_u
Book	K = 5	0.847982	0.794460	0.779996
	K = 10	0.847982	0.791851	0.780546
Movie	K = 5	0.931596	0.816496	0.799368
	K = 10	0.931596	0.813084	0.799911



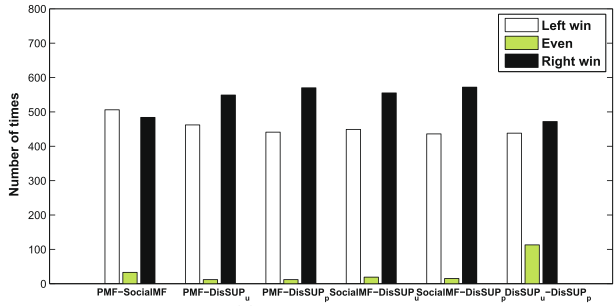
(a) K = 5 on *book*.



(b) K = 10 on *book*.



(c) K = 5 on *movie*.



(d) K = 10 on *movie*.

Figure 4 Paired NDCG comparison for *controversial item recommendation*. The paired comparison methods are shown below the bars and axis. The left bar means the number of times the left method wins (denoted as “Left win”), the middle bar means both methods have the same NDCG value (denoted as “Even”) and the right bar means the number of times the right one wins (denoted as “Right win”)

Table 6 Controversial item recommendation average NDCG

Data set		PMF	SocialMF	DisSUP$_u$	DisSUP$_p$
Book	K = 5	0.944056	0.944519	0.946892	0.947922
	K = 10	0.944517	0.945327	0.946727	0.947039
Movie	K = 5	0.933786	0.934420	0.937950	0.939387
	K = 10	0.934909	0.935309	0.937765	0.938100

least 10 ratings in the training set. Then, based on these ratings we can infer who has the most influence to their followers on the item. The ground truth is the number of their followers that rate on the item in the testing set. The more their followers rate on the item in the testing set, the more influential the user to his followers. We finally select 96 new appearance books and 104 new appearance movies for evaluation according to our training sets.

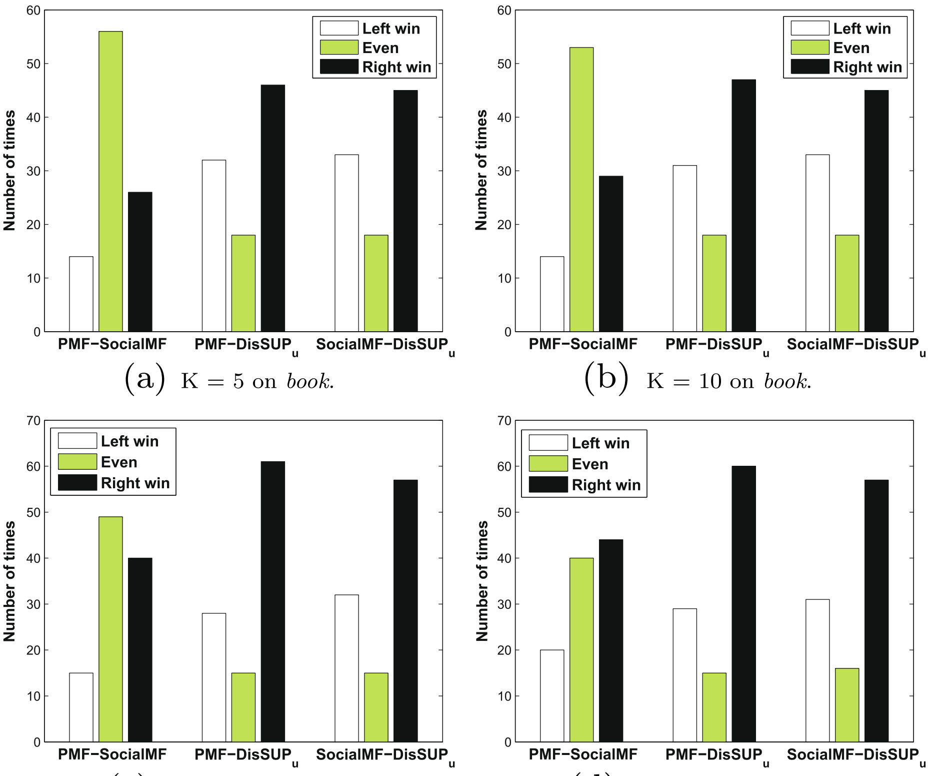
Table 7 shows the average NDCG comparison of the three methods. Our DisSUP$_u$ method outperforms the 2 baseline methods. The results of SocialMF method are slightly better than PMF method, the average improvement on the two data sets with two different settings of K is about 0.002990. Compared to SocialMF, the average improvement of DisSUP$_u$ method is about 0.016534, which is about 5.53 times of the average improvement of the comparison between SocialMF and PMF.

Figure 5 shows us the results of paired NDCG comparison. Generally speaking, we still can get the conclusion that DisSUP$_u$ method is the best of the three and SocialMF is better than PMF in this comparison, as the number of times that DisSUP$_u$ wins is always more than that of the other two methods and SocialMF always wins more than PMF. In addition, the "Even" number of "PMF-SocialMF" is much bigger than both that of "PMF-DisSUP$_u$" and "SocialMF-DisSUP$_u$", which indicates that the incorporation of multi-faceted correlation improves the latent factor model a lot compared to the single correlation.

We summarize the experiments here. As we can see from the above experiments, our DisSUP methods (DisSUP$_u$ and DisSUP$_p$) outperform PMF and SocialMF methods in the three introduced applications. Firstly, the results in *social rating prediction* show that the incorporation of *individual interests* and *implicit multi-faceted social relationship* can improve the learning of *overall interests* of social network users, especially for cold start users. Secondly, the results in *controversial item recommendation* show that the introduced individual interests are more effective in capturing users' preference on controversial items. Finally, experimental results of *1-hop influential user identification* show that the introduced multi-faceted social relationship help to improve the social network users' influence in item level.

Table 7 1-hop influential user identification average NDCG

Data set		PMF	SocialMF	DisSUP$_u$
Book	K = 5	0.869338	0.871953	0.886481
	K = 10	0.869440	0.870816	0.886178
Movie	K = 5	0.840952	0.844438	0.863268
	K = 10	0.84136	0.845841	0.863255



(a) K = 5 on *book*. (b) K = 10 on *book*. (c) K = 5 on *movie*. (d) K = 10 on *movie*.

Figure 5 Paired NDCG comparison for *1-hop influential user identification*. The meanings of the bars are the same with that of Figure 4

6 Related work

In this section, we review some related work on social recommendation. We review some works on recommender systems first. Then we further review works on social recommendation.

The most commonly used technique in recommender systems is collaborative filtering. Two types of collaborative filtering approaches are widely studied, namely the memory-based and model-based methods. The memory-based collaborative approaches include user-based methods [3, 9, 13] and item-based methods [5, 14, 20]. User-based approaches predict the ratings of users based on the rating behaviour similarity between users, and item-based approaches predict the ratings of users based on the computed similarities between the predicting items and those items chosen by the users. In the model-based approaches, predefined models are trained by the training data sets. They include the clustering model [25], aspect model [21] and the latent factor model [18], our method is based on the latent factor model. All the above methods are based on the assumption that users are independent and identically distributed, but in the context of social networks, this assumption does not hold, so researches on recommendation for social network users become necessary. Next we will review some works on social recommendation.

Most social recommendation works are based on trust propagation, they can be categorized into memory-based (e.g., [8, 11, 17]) and model-based (e.g., [12, 15, 16, 22]) methods. TidalTrust was proposed in [8], which performs a modified breadth-first search in the trust network to compute a prediction. MoleTrust in [17] uses a similar idea as TidalTrust, the difference is that MoleTrust considers all raters up to a maximum-depth given as an input and maximum-depth is independent of any specific user and item. In order to consider enough ratings without suffering from noisy data, Jamali and Ester [11] propose a random walk method (TrustWalker) which combines trust-based and item-based recommendation. Ma et al. [16] developed a factor analysis method based on the latent factor model. Ma et al. [15] proposed a method which is a linear combination of basic latent factor approach [18] and a social network based approach. SocialMF proposed in [12] factorizes the rating matrix with the regularization of trust propagation. There are also generative model proposed for making recommendation in social networks. Ye et al. [26] incorporate social influence into generative collaborative filtering model. Chua et al. [4] proposed generative model for item adoptions in social networks.

Indeed our work is different from the previous works on social recommendation in the following aspects. Firstly, most of the previous works only pay attention to the similarity between users and their followees, but lose the sight of their differences. For example, Tang et al. [23] considered local and global social context for recommendation, which only focus on shared interest. Although Feng and Qian [6] investigated similar idea to us that taking individual interest into consideration while making recommendation under the context of social networks, they proposed to use pre-computed the individual interest and single dimensional social influence statically before learning the latent features of users and items. Differently, our method can learn these variables simultaneously during the model training. Secondly, most of previous works (except [22]) just introduce single relationship between users while our method introduces multi-faceted relationship between users. Thirdly, although Tang et al. [22] first introduced the multi-faceted relationship between users, the introduced facets are predefined by humans which is labor intensive and inevitably correlated to each other. However, their method did not take the correlation between the preset facets into consideration. In particular, our method introduces facets in an unsupervised way that we regard each of the latent factors as a facet of the relation between users.

7 Conclusion and future work

In this paper, we revisited the problem of profiling users' interests in the context of social network. We discovered the phenomenon that although users' interests are similar to their followees, there are still differences. Based on this observation, we proposed that social users' *overall interests* can be decomposed into two parts, namely the *individual interests* and the *shared interests*. Besides, we also introduced multi-faceted social relationship in an unsupervised way. Then, we developed a novel **DisSUP** model to capture the user's overall interests and individual interests, and also the unsupervised multi-faceted social relationship between social users. Thirdly, we applied the proposed model into three practical applications, namely *social rating prediction*, *controversial item recommendation* and *1-hop influential user identification*. Finally, experiments on two large-scale real-world datasets from Douban clearly validate that our proposed model outperforms other baselines in terms of the three applications.

For future work, there are several new research directions worth to be investigated. Firstly, the proposed model does not take the temporal information into consideration, we are going to study the interest drifting in the future work. Secondly, the information on user profile and item profile is not used in this study. How to incorporate the profile information into our proposed model is an open direction for future work. Finally, the correlation of multi-faceted social interests relationship and social influence is worth to be further investigated.

Acknowledgments This research was partially supported by grants from the National Science Foundation for Distinguished Young Scholars of China (Grant No. 61325010), the Natural Science Foundation of China (Grant No.s 61403358 and 71329201), the Science and Technology Program for Public Wellbeing (Grant No. 2013GS340302), and the National High Technology Research and Development Program of China (Grant No. 2014AA015203). Ping was supported by the National Natural Science Foundation of China (No.61473274), and National High Technology Research and Development Program of China (No.2014AA015105).

Appendix

Before conducting the experiment methods, we pre-processed the ratings as follows: let $LRate$ and $HRate$ denote the Lowest and the Highest rating in the training set, then

- $r^1_{u,v} = \frac{r_{u,v} - LRate}{HRate - LRate}$;
- $r^2_{u,v} = r^1_{u,v} - r^1$, where $r^1 = \frac{1}{|\Omega|} \sum_{\Omega} r^1_{u,v}$;
- let $\{r^2_{u,v}\}$ be the input of the methods.

When we apply the trained models to the applications, we recover the predicted values as follows:

$$\tilde{r}^1_{u,v} = \left\{ \begin{array}{c} 0,\ \tilde{r}^2_{u,v} < 0, \\ 1,\ \tilde{r}^2_{u,v} > 1, \\ \tilde{r}^2_{u,v},\ else. \end{array} \right\} \tilde{r}^2_{u,v} = r^1_{u,v} + \mathbf{u} \cdot \mathbf{v}. \tag{26}$$

$$\tilde{r}_{u,v} = LRate + \tilde{r}^1_{u,v}(HRate - LRate) \tag{27}$$

$$\varphi_{u,v} = \left\{ \begin{array}{c} 0,\ \varphi^1_{u,v} < 0, \\ L_u,\ \varphi^1_{u,v} > L_u, \\ \varphi^1_{u,v},\ else. \end{array} \right. \tag{28}$$

where $\varphi^1_{u,v} = \rho_u \cdot \mathbf{v} + L_u \cdot r^1$, $L_u = \sum_{u' \in \mathcal{D}(u)} \frac{||\mathbf{w}_{u',u}||_1}{K}$ ($||\cdot||_1$ means the l_1 norm of vectors).

$$\eta_{u,v} = \left\{ \begin{array}{c} 0,\ \eta^1_{u,v} < 0, \\ T_u,\ \eta^1_{u,v} > T_u, \\ \eta^1_{u,v},\ else. \end{array} \right. \tag{29}$$

and $\eta^1_{u,v} = \mathbf{t}_u \cdot \mathbf{v} + T_u \cdot r^1$, $T_u = \sum_{u' \in \mathcal{D}(u)} T_{u',u}$.

$$\theta_{u,v} = \left\{ \begin{array}{c} 0,\ \theta^1_{u,v} < 0, \\ P_u,\ \theta^1_{u,v} > P_u, \\ \theta^1_{u,v},\ else. \end{array} \right. \tag{30}$$

where $\theta^1_{u,v} = \mathbf{p}_u \cdot \mathbf{v} + P_u \cdot r^1$, $P_u = \sum_{u' \in \mathcal{F}(u)} \frac{\mathbf{h}_{u,u'} \cdot \mathbf{w}_{u,u'}}{K}$.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
2. Boyd, S.P., Vandenberghe, L.: Convex optimization. Cambridge University Press (2004)
3. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: UAI '98, pp. 43–52 (1998)
4. Chua, F.C.T., Lauw, H.W., Lim, E.P.: Generative models for item adoptions using social correlation. IEEE Trans. Knowl. Data Eng. **25**(9), 2036–2048 (2013)
5. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inf. Syst. **22**(1), 143–177 (2004)
6. Feng, H., Qian, X.: Recommendation via user's personality and social contextual. In: CIKM '13, pp. 1521–1524. ACM (2013)
7. Friedkin, N.E.: A structural theory of social influence, vol. 13. Cambridge University Press (2006)
8. Golbeck, J.A.: Computing and applying trust in Web-based social networks (2005)
9. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR '99, pp. 230–237 (1999)
10. Hofmann, T.: Latent semantic models for collaborative filtering. ACM Trans. Inf. Syst. **22**, 89–115 (2004)
11. Jamali, M., Ester, M.: Trustwalker: a random walk model for combining trust-based and item-based recommendation. In: KDD '09, pp. 397–406 (2009)
12. Jamali, M., Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks. In: RecSys '10, pp. 135–142 (2010)
13. Jin, R., Chai, J.Y., Si, L.: An automatic weighting scheme for collaborative filtering. In: SIGIR '04, pp. 337–344 (2004)
14. Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet Comput. **7**(1), 76–80 (2003)
15. Ma, H., King, I., Lyu, M.R.: Learning to recommend with social trust ensemble. In: SIGIR '09, pp. 203–210 (2009)
16. Ma, H., Yang, H., Lyu, M.R., King, I.: Sorec: social recommendation using probabilistic matrix factorization. In: CIKM '08, pp. 931–940 (2008)
17. Massa, P., Avesani, P.: Trust-aware recommender systems. In: RecSys '07, pp. 17–24 (2007)
18. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: NIPS '07, pp. 1257–1264 (2007)
19. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted boltzmann machines for collaborative filtering. In: ICML '07, pp. 791–798 (2007)
20. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW '01, pp. 285–295 (2001)
21. Si, L., Jin, R.: Flexible mixture model for collaborative filtering. In: ICML '03, pp. 704–711 (2003)
22. Tang, J., Gao, H., Liu, H.: Mtrust: discerning multi-faceted trust in a connected world. In: WSDM '12, pp. 93–102 (2012)
23. Tang, J., Hu, X., Gao, H., Liu, H.: Exploiting local and global social context for recommendation. In: Proceedings of the twenty-third international joint conference on artificial intelligence, pp. 2712–2718. AAAI Press (2013)
24. Wang, Y., Liwei, W., Li, Y., He, D., Chen, W., Liu, T.Y.: A theoretical analysis of ndcg ranking measures. In: 26th annual conference on learning theory (2013)
25. Xue, G.R., Lin, C., Yang, Q., Xi, W., Zeng, H.J., Yu, Y., Chen, Z.: Scalable collaborative filtering using cluster-based smoothing. In: SIGIR '05, pp. 114–121 (2005)
26. Ye, M., Liu, X., Lee, W.C.: Exploring social influence for recommendation: a generative model approach. In: SIGIR '12, pp. 671–680. ACM (2012)