Integrating Community Context Information into a Reliably Weighted Collaborative Filtering System Using Soft Ratings

Van-Doan Nguyen, Van-Nam Huynh, *Member, IEEE,* Songsak Sriboonchitta

***Abstract*—In this paper, we aim at developing a new collaborative filtering recommender system using soft ratings, which is capable of dealing with both imperfect information about user preferences and the sparsity problem. On the one hand, Dempster-Shafer theory is employed for handling the imperfect information due to its advantage in providing not only a flexible framework for modeling uncertain, imprecise, and incomplete information, but also powerful operations for fusion of information from multiple sources. On the other hand, in dealing with the sparsity problem, community context information that is extracted from the social network containing all users is used for predicting unprovided ratings. As predicted ratings are not a hundred percent accurate, while the provided ratings are actually evaluated by users, we also develop a new method for calculating user-user similarities, in which provided ratings are considered to be more significant than predicted ones. In the experiments, the developed recommender system is tested on two different data sets; and the experiment results indicate that this system is more effective than CoFiDS, a typical collaborative filtering recommender system offering soft ratings.**

***Index Terms*—Recommender Systems, Uncertain Reasoning, Dempster-Shafer Theory.**

I. Introduction

Over the last two decades, recommender systems [3], [4], [5], [6], [7] have been developed and widely applied into e-commerce applications [8], [9], [10]. However, most of previous studies on recommendation techniques have unfortunately neglected the important issue of imperfections which may be caused due to ambiguities and uncertainties in user ratings [11]. Such imperfect ratings need to be appropriately represented and processed so as to improve quality and reliability of recommender systems [12]. Over the years, a number of theories have been developed for the purpose of modeling imperfect information, such as probability theory [13], Dempster-Shafer theory (DST) [14], [15], possibility theory [16], and so on. Among these, DST is one of the most general theories in which various kinds of imperfect information can be represented [11].

Typically, in conventional recommender systems, the rating domain is often defined as a totally ordered finite set of rating scores used by users as an instrument to express their preferences on items. It is also common for users to use only one rating score (called hard rating) for each item. However, due to the subjective and qualitative nature of user preferences, in practice there may be users who are hesitant to use "hard rating" when evaluating items. For example, in a recommender system offering a rating domain $\Theta = \{\theta_1, \theta_2, ..., \theta_L\}$, according to some aspects, a user intends to rate an item as θ_i, while regrading other aspects, this user also wants to rate the item by θ_{i+1}. Moreover, in some scenarios, the use of hard ratings may be not appropriate; for instance, assume that a user has rated two items I_1 and I_3 as θ_i and θ_{i+1} respectively, then it would be difficult for the user to precisely score some item I_2 that he/she considers better than item I_1 but worse than item I_3. In such situations, a combination of rating scores such as $\{\theta_i, \theta_{i+1}\}$ (called soft ratings, being defined as subsets or interval score in the rating domain) could be used more comfortably and confidently. With this observation, soft ratings have been recently introduced to capture the subjective and qualitative nature of user preferences in recommender systems.

According to the previous studies, recommender systems using soft ratings are developed based on DST. Thus, using soft ratings can be considered to be a new perspective to model not only subjective and qualitative but also imperfect information about user preferences. DST also supports modeling missing data by the vacuous mass structure and generating both hard and soft decisions as the recommendations presented by singletons and composites, respectively. Specially, with Dempster's rule of combination [14], information about user preferences from different sources can be combined easily.

Furthermore, in the research area of recommender systems, collaborative filtering is the most popular technique [17], and the sparsity problem is considered to be a major drawback affecting the quality of recommendations in collaborative filtering recommender systems [18]. Moreover, as we can observe, recently social networks are growing very fast and playing a significant role on the Internet. Naturally, in social networks, users form into communities whose members interact frequently with one another [19]; and when consulting for advice before buying a new item, people tend to believe in recommendations from the members in the same community rather than recommendations from anonymous users. Indeed, social networks contain a huge amount of information that could be valuable for dealing with the sparsity problem as well as improving the quality of recommendations.

In this paper, we develop a novel collaborative filtering recommender system which is capable of dealing with imperfect

V.D. Nguyen, V.N. Huynh are with the Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923-1292, Japan. Email: {nvdoan, huynh}@jaist.ac.jp.

S. Sriboonchitta is with Chiang Mai University, 50200 Chiang Mai, Thailand, Email: songsakecon@gmail.com

This paper is a significantly extended and combined version of two conference papers presented at PRICAI-2014 [1] and ECSQARU-2015 [2].

information as well as the sparsity problem. In this system, user preferences are modeled by using DST, and community context information extracted from the social network containing all users, is employed for predicting unprovided ratings. In short, this paper contains two main contributions: (1) a new approach to overcome the sparsity problem by using community context information, and (2) a new method for computing user-user similarities, in which provided ratings are weighted more important than predicted ratings. Moreover, the experiment results indicate that using community context information and assigning weights to ratings help to improve the quality of recommendations.

The remainder of this paper is organized as follows. Section II presents DST, some well-known techniques for detecting communities in a social network, and related work. Then, Section III describes details of the proposed system and Section IV provides the system implementation and discussion. Finally, Section V wraps up the paper with some concluding remarks.

II. Background and Related Work

A. Basics of Dempster-Shafer Theory

DST [14], [15] is one of the popular theories for modeling uncertain, imprecise, and incomplete information. Let us consider a problem domain which is represented by a finite set $\Theta = \{\theta_1, \theta_2, ..., \theta_L\}$ of exhaustive and mutually exclusive hypotheses [15]. A function $m : 2^\Theta \rightarrow [0, 1]$ is called a mass function or basic probability assignment (BPA) if it satisfies $m(\emptyset) = 0$ and $\sum_{A \subseteq \Theta} m(A) = 1$. Mass function m is vacuous if $m(\Theta) = 1$ and $\forall A \neq \Theta, m(A) = 0$. A subset $A \subseteq \Theta$ with $m(A) > 0$ is called a focal element of mass function m.

According to Smets' two-level view in the so-called transferable belief model [20], [21], when a decision needs to be made, the mass function m should be transformed into a probability distribution called pignistic probability function $Bp : \Theta \rightarrow [0, 1]$ defined by

$$Bp(\theta_i) = \sum_{\{A \subseteq \Theta | \theta_i \in A\}} \frac{m(A)}{| A |}$$

In the context of DST, two useful operations which play a significant role are known as Dempster's rule of combination and discounting [15]. Let us consider two mass functions m_1 and m_2 defined on the same problem domain Θ. Dempster's rule of combination, denoted by $m = m_1 \oplus m_2$, can be used for combining these two mass functions as follows

$$\begin{aligned} m(\emptyset) &= 0; \\ m(A) = \frac{1}{1 - K} \sum_{\{B, C \subseteq \Theta | B \cap C = A\}} & m_1(B) m_2(C); \\ \text{where } K = \sum_{\{B, C \subseteq \Theta | B \cap C = \emptyset\}} & m_1(B) m_2(C) \neq 0. \end{aligned}$$

Here, K represents the basic probability mass associated with conflict. As remarked in the literature, Dempster's rule of combination serves as a powerful tool for fusing information from different sources [22].

The discounting operation is used when the information source providing mass function m has probability δ of reliability. In this case, one may adopt $1 - \delta \in [0, 1]$ as one's discount rate, resulting in a new mass function m^δ defined by

$$m^\delta(A) = \begin{cases} \delta m(A), & \text{for } A \subset \Theta; \\ \delta m(\Theta) + (1 - \delta), & \text{for } A = \Theta. \end{cases}$$

B. Community Detection

Basically, a social network can be considered to be a social structure made of nodes regarding individuals or organizations, and edges that connect nodes in various relations such as friendship or kinship [19]. Additionally, this network is usually represented by a graph or an adjacency matrix.

According to the previous studies, numerous techniques have been introduced for identifying communities in social networks, such as mimicking human pair-wise communication [23], [24], removal of high-betweenness [25], [26], [27], detection of dense sub-graphs [28], and modularity optimization [29], [30]. Typically, in a social network, an individual can belong to a variety of communities, therefore, the communities are usually overlapping. In most previous studies, algorithms to detect overlapping communities can be classified into five main categories as clique percolation, local expansion and optimization, line graph and link partitioning, fuzzy detection, and agent-based and dynamical algorithms [31].

In this paper, we adopt Speaker-Listener Label Propagation (SLPA) algorithm [23] for naturally uncovering communities in the social network. The reason is that this algorithm can (1) effectively detect overlapping communities in a large-scale network with the time complexity that is proportional to the number of edges in linear form and (2) avoid producing very small size communities.

C. Related Work

Regarding the literature, recommendation techniques are classified into three main categories: collaborative filtering, content-based, and hybrid [3]. Among these, collaborative filtering, which aims at recommending to a user a list of items other users with similar tastes liked in the past, is widely implemented in e-commerce applications [17], [32]. However, collaborative filtering technique has its own limitations such as the new user problem, the new item problem, and the sparsity problem [3]. The new user issue occurs when a system can not discover preferences of a user because this user has rated none or not enough items. When an item is not rated due to some reasons, for example this item has just been added, the system can not recommend it; in this case the new item issue occurs. The sparsity problem takes place when the number of ratings recorded is very small compared to the number of missing ratings. Among these three limitations, the sparsity problem significantly affects the quality of recommendations in collaborative filtering recommender systems [18].

Over the years, many researchers have focused on tackling the sparsity problem in collaborative filtering recommender systems. The challenge of this problem is how to generate effective recommendations to each specific user when a small number of provided ratings are available; and so far, a variety of methods have been developed for dealing with this problem. According to the literature, matrix factorization techniques [33], [34], [35], [36], [37] have become popular due to the ability to combine good scalability with predictive accuracy; and importantly, these techniques are known as a good choice for implementing recommender systems dealing with the sparsity problem. Some authors suggested to combine collaborative filtering with content-based techniques [38] or employ information from other sources, such as demographic information [39] or implicit preferences derived from users' behaviors [40], for tackling the problem. Recently, integrating recommender systems with social networks has emerged as an active research topic [41], [42], [43], [44], [45]. Up to now, a variety of collaborative filtering recommender systems have been developed based on social networks [46], [47], [48], [49], [50], [51]; and most of these systems employ social trust [52], [53] for overcoming the sparsity problem. Also, some authors have applied matrix factorization techniques for generating recommendations in social networks [54], [55]. However, matrix factorization and the other methods can be only applied into collaborative filtering recommender systems which offer hard ratings.

Additionally, handling imperfect information has been made much progress in the previous studies [56], [57], [58], [59], [60]. In [11], the authors proposed a new collaborative filtering recommender system, called CoFiDS, which overcomes the sparsity problem by employing context information for generating unprovided ratings and deals with imperfect information by using DST for modeling ratings. Also, CoFiDS is a recommender system that offers soft ratings. However, in CoFiDS, the role of predicted ratings is considered to be the same as that of provided ratings, and this system is not able to predict all unprovided ratings as it is expected (see the illustration in Example 1 in Section IV).

The collaborative filtering recommender system proposed in this paper is capable of offering soft ratings and integrating with social network containing all users. Besides, this system overcomes the weaknesses of CoFiDS.

III. Proposed System

A. Data Modeling

Let $\mathbf{U} = \{U_1, U_2, ..., U_M\}$ be the set of all users, and assume that all users join in a social network that is represented by friend relationships. Also, let $\mathbf{I} = \{I_1, I_2, ..., I_N\}$ be the set of all items. Note that in real-world applications, the number of elements in $\mathbf{U}$ and $\mathbf{I}$ can be very large. Users can express their preferences on items by using soft ratings that span over a rating domain containing L preference labels, $\Theta = \{\theta_1, \theta_2, ..., \theta_L\}$. The ratings of all users are represented by a rating matrix $\mathbf{R} = \{r_{i,k}\}$ with $r_{i,k} : 2^\Theta \to [0, 1]$ is the rating of user U_i on item I_k. Each unprovided rating can be modeled by vacuous and considered as manifest lack of evidence. However, it can be seen that vacuous representation is high uncertainty. Thus, community context information is employed for predicting unprovided ratings in order to reduce uncertainty introduced by vacuous.

As remarked in [11], context information that might influence user preferences on items can be considered as concepts for grouping users or items. For instance, in a movie recommender system, characteristics such as genre, user gender, and user occupation are to be regarded as concepts. A concept can contain various groups, e.g. genre consists of several groups such as animation, mystery, action, and comedy.

Supposing that, the proposed system contains P characteristics which can be considered as concepts; and each concept C_p with $1 \le p \le P$ consists of a maximum of Q_p groups. Consequently, context information, denoted by $\mathbf{C}$, is represented as below

$$\mathbf{C} = \{C_1, C_2, ..., C_P\};$$
$$\text{where } C_p = \{G_{p,1}, G_{p,2}, ..., G_{p,Q_p}\} \text{ for } C_p \in \mathbf{C}.$$

Each user U_i can be interested in multiple groups in a given concept $C_p \in \mathbf{C}$, and these groups are identified by mapping functions f_p as below

$$\begin{aligned} f_p : \mathbf{U} &\to 2^{C_p} \\ U_i &\mapsto f_p(U_i) \subseteq C_p. \end{aligned}$$

Similarly, each item I_k can belong to several groups in concept C_p, and these groups are determined by mapping function g_p as follows

$$\begin{aligned} g_p : \mathbf{I} &\to 2^{C_p} \\ I_k &\mapsto g_p(I_k) \subseteq C_p. \end{aligned}$$

In general, the process of recommendations in the proposed system is depicted in Fig. 1. As we can observe in this figure, first, overlapping communities in the social network are detected, and then assume that we achieve a set containing V overlapping communities denoted by $\boldsymbol{\mathcal{C}} = \{\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_V\}$. Second, tasks such as predicting unprovided ratings, computing user-user similarities, selecting neighborhoods and estimating ratings for active users on unseen items are performed in each community independently. Finally, the estimated ratings of each active user in the communities to which he/she belongs are combined, and then suitable recommendations to this user are generated mainly based on the combined results.

Details of tasks in the process of recommendations will be described in the rest of this section,

B. Detecting Overlapping Communities

As mentioned earlier, we employ SLPA algorithm [23] for detecting overlapping communities in the social network. According to this algorithm, each node holds and handles a memory of the labels received from other nodes. Briefly, the SLPA algorithm consists of three stages [23], as shown below

1) First, the memory of every node is initialized with a unique label.
2) Next, the following communication steps are repeated until the maximum iteration T is reached:
   - One node is chosen to be a listener.
   - Each neighbor of the listener chooses a label with probability proportional to the occurrence frequency in its memory and sends the label to the listener.

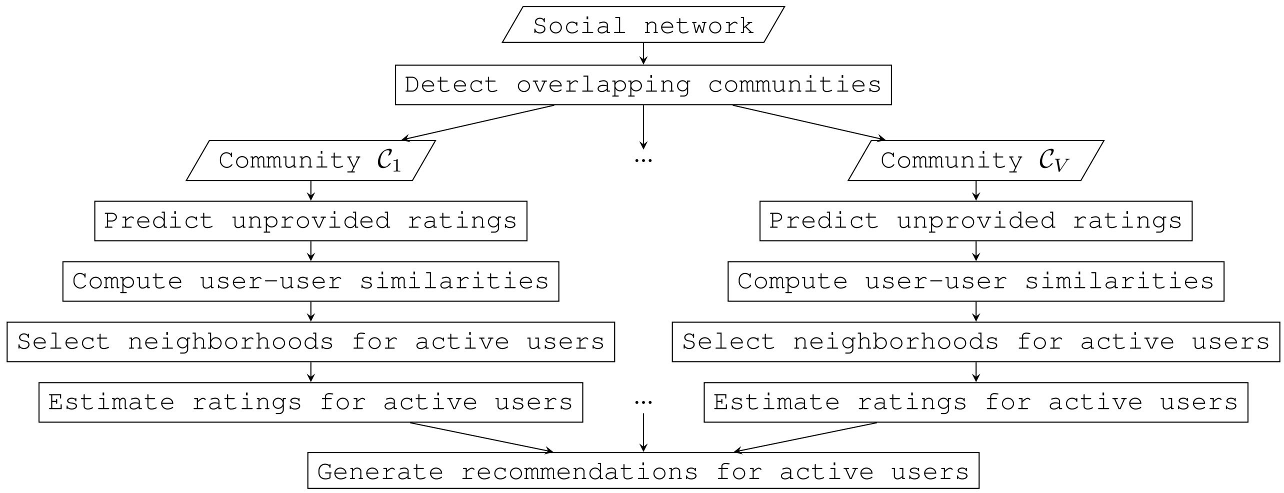

Fig. 1: The process of recommendations (adapted from [61])

- The listener selects the most popular label received and adds the label to its memory.

3) Finally, overlapping communities are identified based on the labels in the memories of nodes.

Note that some communities detected by using SLPA algorithm might contain a very large or small number of users. Therefore, we continue using this algorithm for dividing the very large communities into smaller communities (if possible) and we assign each member in the very small communities to the community that contains most of its neighbors. Formally, let $\mathtt{M}_{min}$ and $\mathtt{M}_{max}$ be the minimum and maximum number of users in a community which we expect to achieve. Communities consisting of more than $\mathtt{M}_{max}$ or less than $\mathtt{M}_{min}$ users are considered as very large or small communities, respectively. The values of $\mathtt{M}_{min}$ and $\mathtt{M}_{max}$ can be selected according to each specific application.

We assume that, after executing SLPA algorithm, we get V overlapping communities in total. Rating matrix $\mathbf{R}$ is then separated into V sub-rating matrixes, denoted by $\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_V$, corresponding to these communities. Each sub-rating matrix $\mathbf{R}_v$ consists of the ratings of all members in community $\mathcal{C}_v$.

C. Performing on Communities

Note that the tasks described in this subsection are performed in each community $\mathcal{C}_v \in \mathbf{C}$ independently.

1) Generating Unprovided Ratings: Let ${}^R\mathbf{I}_{i,v}$ and ${}^R\mathbf{U}_{k,v}$ denote the set of items which have been rated by user U_i and the set of users who have rated item I_k, respectively. These sets can be defined as follows

$$ {}^R\mathbf{I}_{i,v} = \{I_l \in \mathbf{I} \mid r_{i,l,v} \neq \text{vacuous}\}; $$
$$ {}^R\mathbf{U}_{k,v} = \{U_l \in \mathcal{C}_v \mid r_{l,k,v} \neq \text{vacuous}\}. $$

For a given concept C_p with $p = \overline{1,P}$, the group preference of all users on item I_k regarding each group $G_{p,q} \in g_p(I_k)$, with $1 \leq q \leq Q_p$, defined by ${}^Gm_{p,q,k,v} : 2^\Theta \to [0,1]$, is obtained by combining provided ratings of users who are also interested in group $G_{p,q}$, as below

$$ {}^Gm_{p,q,k,v} = \bigoplus_{\{i \mid I_k \in {}^R\mathbf{I}_{i,v}, G_{p,q} \in f_p(U_i), G_{p,q} \in g_p(I_k)\}} r_{i,k,v}. \quad (1) $$

Suppose that the rating $r_{i,k,v}$ corresponsing to the evaluation of user U_i on item I_k is not provided. Unprovided rating $r_{i,k,v}$ is generated mainly based on the method suggested in [11], as follows

- First, since user U_i belongs to community $\mathcal{C}_v$, U_i's preference on item I_k is influenced by the preferences of members in this community. Additionally, it would appear that users who are interested in the same group of a given concept can be expected to possess similar preferences regarding this group. Under such an observation, for each concept C_p, if user U_i is interested in group $G_{p,q} \in C_p$ then U_i's preference on item I_k regarding group $G_{p,q}$, denoted by ${}^Gm_{i,p,q,k,v} : 2^\Theta \to [0,1]$, can be assigned the group preference of users in community $\mathcal{C}_v$ on item I_k regarding group $G_{p,q}$, as follows

$$ {}^Gm_{i,p,q,k,v} = {}^Gm_{p,q,k,v}. \quad (2) $$

- Second, U_i's concept preference on item I_k regarding concept C_p, denoted by ${}^Cm_{i,p,k,v} : 2^\Theta \to [0,1]$, is achieved by combining all related group preferences of user U_i on item I_k, as shown below

$$ {}^Cm_{i,p,k,v} = \bigoplus_{\{q \mid G_{p,q} \in f_p(U_i), G_{p,q} \in g_p(I_k)\}} {}^Gm_{i,p,q,k,v}. \quad (3) $$

- Next, U_i's context preference on item I_k, denoted by ${}^{\mathbf{C}}m_{i,k,v} : 2^\Theta \to [0,1]$, is computed by combining all related concept preferences of user U_i on item I_k, as follows

$$ {}^{\mathbf{C}}m_{i,k,v} = \bigoplus_{p=\overline{1,P}} {}^Cm_{i,p,k,v}. \quad (4) $$

- Finally, if user U_i's context preference on item I_k is vacuous, such as the illustration in Example 1 in Section IV, the unprovided rating $r_{i,k,v}$ is assigned the result obtained by combining the existing ratings on item I_k as shown below

$$ r_{i,k,v} = \bigoplus_{\{j \mid U_j \in {}^R\mathbf{U}_{k,v}\}} r_{j,k,v}. \quad (5) $$

TABLE I: Reliability function $\mu(x_{i,k}, x_{j,k})$

$x_{i,k}$	$x_{j,k}$	$\mu(x_{i,k}, x_{j,k})$
0	0	1
0	1	$1-w_1$
1	0	$1-w_1$
1	1	$1-2\times w_1 - w_2$

Otherwise, unprovided rating $r_{i,k,v}$ is assigned the U_i's context preference, as follows

$$r_{i,k,v} = {}^{\mathbf{C}}m_{i,k,v}. \quad (6)$$

Please note that, at this point, except for new items, all unprovided ratings in sub-rating matrix R_v are predicted.

2) Computing User-User Similarities: In sub-rating matrix $\mathbf{R}_v = \{r_{i,k,v}\}$, each entry $r_{i,k,v}$ represents the preference of user U_i on a single item I_k, $r_{i,k,v} = r_{i,k}$. The preference of this user toward all items as one can be captured from the cross-product $\mathbf{\Theta}_v = \Theta_1 \times \Theta_2 \times ... \times \Theta_N$, where $\Theta_i = \Theta$ with $i = \overline{1,N}$ [11], [62].

Let $F_{i,k,v}$ denote the focal set of $r_{i,k,v}$. The cylindrical extension of each focal element $A \in F_{i,k,v}$ to the cross-product $\mathbf{\Theta}_v$ is $cyl_{\mathbf{\Theta}_v}(A) = [\Theta_1...\Theta_{i-1}A\Theta_{i+1}...\Theta_N]$; and the mapping $M_{i,k,v} : 2^{\mathbf{\Theta}_v} \to [0,1]$ with $M_{i,k,v}(B) = r_{i,k,v}(A)$ for $B = cyl_{\mathbf{\Theta}_v}(A)$, and 0 otherwise, is a valid mass function defined on $\mathbf{\Theta}_v$ [62]. In [11], the authors have pointed out that the preference of user U_i toward all items as a whole is represented by mass function $M_{i,v} : 2^{\mathbf{\Theta}_v} \to [0,1]$ with $M_{i,v} = \bigoplus_{k=1}^{N} M_{i,k,v}$. Additionally, the pignistic probability distribution according to mass function $M_{i,v}$, denoted by $Bp_{i,v}$, is computed as below

$$Bp_{i,v}\left(\theta_{i_1} \times ... \times \theta_{i_N}\right) = \prod_{k=1}^{N} Bp_{i,k,v}(\theta_{i_k}),$$

where $Bp_{i,k,v}$ is the pignistic probability distribution corresponding to the preference of user U_i on item I_k [11].

Let us consider two users U_i and U_j. The preferences of these users toward all items as a whole are $M_{i,v}$ and $M_{j,v}$, respectively. As remarked in [11], the distance between $M_{i,v}$ and $M_{j,v}$ is $D_v(M_{i,v}, M_{j,v}) = CD(Bp_{i,v}, Bp_{j,v})$, where $Bp_{i,v}$ and $Bp_{j,v}$ are the pignistic probability distributions according to $M_{i,v}$ and $M_{j,v}$, respectively, and $CD()$ is the Chan-Darwiche (CD) distance measure [63] defined by

$$CD(Bp_{i,v}, Bp_{j,v}) = \ln \max_{\boldsymbol{\theta}_i \in \mathbf{\Theta}_v} \frac{Bp_{j,v}(\boldsymbol{\theta}_i)}{Bp_{i,v}(\boldsymbol{\theta}_i)} - \ln \min_{\boldsymbol{\theta}_i \in \mathbf{\Theta}_v} \frac{Bp_{j,v}(\boldsymbol{\theta}_i)}{Bp_{i,v}(\boldsymbol{\theta}_i)}.$$

Also, according to the result that is proven in [11], the distance between $M_{i,v}$ and $M_{j,v}$ is

$$D_v(M_{i,v}, M_{j,v}) = \sum_{k=1}^{N} CD(Bp_{i,k,v}, Bp_{j,k,v}), \quad (7)$$

where $Bp_{i,k,v}$ and $Bp_{j,k,v}$ are pignistic probability distributions corresponding to preferences of two users U_i and U_j on item I_k, respectively.

As we can see in equation (7), the role of each expression $CD(Bp_{i,k,v}, Bp_{j,k,v})$ is considered to be the same for all items regardless of whether ratings on them are predicted or provided. Let us consider the ratings $r_{i,k,v}$ and $r_{j,k,v}$ corresponding to the evaluations of users U_i and U_j on item I_k, respectively. Since $Bp_{i,k,v}$ and $Bp_{j,k,v}$ are derived from ratings $r_{i,k,v}$ and $r_{j,k,v}$, the expression $CD(Bp_{i,k,v}, Bp_{j,k,v})$ is only fully reliable when both $r_{i,k,v}$ and ${}_{j,k,v}$ are provided ratings. Otherwise, when at least one of the two ratings is predicted, the expression is not fully reliable.

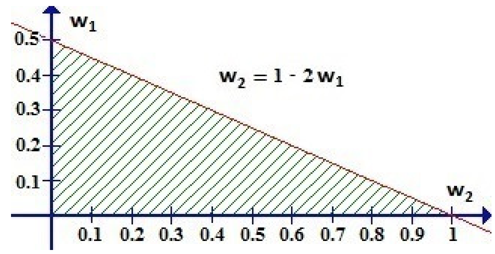

Fig. 2: Domains of two reliability coefficients

For improving the accuracy of measuring the distance between two users, provided ratings should be considered more important than predicted ones. To achieve this goal, we propose a new method for measuring the distance between $M_{i,v}$ and $M_{j,v}$, as follows

$$\hat{D}_v(M_{i,v}, M_{j,v}) = \sum_{k=1}^{N} \mu(x_{i,k}, x_{j,k}) CD(Bp_{i,k,v}, Bp_{j,k,v}), \quad (8)$$

where $\mu(x_{i,k}, x_{j,k}) \in [0,1]$ is a reliability function that represents the trust of the evaluations of both users U_i and U_j on item I_k. $\forall (i,k), x_{i,k} \in \{0,1\}$, when $r_{i,k,v}$ is a predicted rating, $x_{i,k} = 0$; otherwise, $x_{i,k} = 1$.

Note that, the distinguishing of predicted and provided ratings will not destroy the elegance of the Chan-Darwiche distance measure because reliability function $\mu(x_{i,k}, x_{j,k}) \in [0,1]$. $\mu(x_{i,k}, x_{i,k}) = 1$ when both ratings $r_{i,k,v}$ and $r_{j,k,v}$ are predicted; $\mu(x_{i,k}, x_{i,k}) < 1$ when at least one of the ratings is provided, and it means that user U_i has a higher opportunity to be selected as a member in the neighborhood set of user U_j and vice versa.

In practice, the reliability function $\mu(x_{i,k}, x_{j,k})$ should be defined according to each specific application. Generally, we suggest a reliability function, as below

$$\mu(x_{i,k}, x_{j,k}) = 1 - w_1(x_{i,k} + x_{j,k}) - w_2 x_{i,k} x_{j,k}, \quad (9)$$

where $w_1 \geq 0$ and $w_2 \geq 0$ are two reliability coefficients which represent the state when one of two ratings $r_{i,k,v}$ and $r_{j,k,v}$ is provided and both of the two ratings are provided ratings, respectively.

Since $x_{i,k}, x_{j,k} \in \{0,1\}$, the function $\mu(x_{i,k}, x_{j,k})$ has four different cases, as illustrated in Table I. Under the condition $0 \leq \mu(x_{i,k}, x_{j,k}) \leq 1$, the domains of two reliability coefficients w_1 and w_2 are illustrated in the parallel diagonal line shading area in Fig. 2.

Let us consider a monotonically decreasing function ψ: $[0,\infty] \to [0,1]$ satisfying $\psi(\infty) = 0$ and $\psi(0) = 1$. With respect to this function, the user-user similarity between two users U_i and U_j is computed, as follows

$$s_{i,j,v} = \psi(\hat{D}_v(M_{i,k}, M_{j,k})).$$

To compute user-user similarities in the proposed system, we select the monotonically decreasing function as below

$$\psi(x) = e^{-\gamma \times x} \text{ with } \gamma \in (0, \infty). \tag{10}$$

Additionally, the user-user similarities among all users in community $\mathcal{C}_v$ are represented in a similarity matrix $\mathbf{S}_v = \{s_{i,j,v}\}$ with $U_i, U_j \in \mathcal{C}_v$.

Similar to the CoFiDS system developed in [11], the highest computational effort of the proposed system lies in determining the similarity between two users. According to Equation (9), the complexity of the reliability function is $O(1)$. Consequently, the complexity for determining the distance between two users in the proposed system (Equation (8)) is the same as that in the predecessor (Equation (7)). As shown in [11], the complexity for determining the distance between two users is $O(LN)$ where L is the number of preference labels and N is the number of items. So, in the general case, the complexity for determining the distance between two users in the proposed system is $O(LN)$.

3) Selecting Neighborhoods: We adopt the method introduced in [64] for determining neighborhoods for active users in the proposed system. This method is effective because of the inclusion of two popular strategies known as K-nearest neighbor and minimum similarity shareholding. Let $\mathcal{N}_{i,k,v}$ denote the neighborhood set of user U_i regarding item I_k. To obtain the set $\mathcal{N}_{i,k,v}$, other users who have rated item I_k and whose user-user similarities with user U_i are greater than or equal to a threshold τ are extracted; after that, top K users with the highest user-user similarities are selected from the extracted list.

4) Estimating Ratings: After obtaining neighborhood set $\mathcal{N}_{i,k,v}$, the rating $r_{j,k,v}$ of each neighbor $U_j \in \mathcal{N}_{i,k,v}$ is discounted by the user-user similarity $s_{i,j,v} \in \mathbf{S}_v$ between user U_i and U_j as follows

$$r_{j,k,v}^{s_{i,j,v}}(A) = \begin{cases} s_{i,j,v} r_{j,k,v}(A), & \text{for } A \subset \Theta; \\ s_{i,j,v} r_{j,k,v}(\Theta) + (1 - s_{i,j,v}), & \text{for } A = \Theta. \end{cases}$$

The estimated rating of user U_i on unrated item I_k is computed as below

$$\hat{r}_{i,k,v} = r_{i,k,v} \oplus \bar{r}_{i,k,v}, \quad \text{where } \bar{r}_{i,k,v} = \bigoplus_{\{j | U_j \in \mathcal{N}_{i,k,v}\}} r_{j,k,v}^{s_{i,j,v}}. \tag{11}$$

D. Generating Recommendations

Suitable recommendations for each active user U_i are generated according to the number of communities to which this user belongs. In case user U_i is a member of only one community $\mathcal{C}_v$, the last estimated rating of this user on item I_k, denoted by $\hat{r}_{i,k}$, is achieved as follows

$$\hat{r}_{i,k} = \hat{r}_{i,k,v}, \tag{12}$$

where $U_i \in \mathcal{C}_v$ and $\hat{m}_{v,i,k}$ is user U_i's estimated rating of user U_i on item I_k in community $\mathcal{C}_v$. If user U_i simultaneously belongs to several communities, the last estimated rating of the user on item I_k is obtained by combining the estimated ratings on item I_k in the communities to which this user belongs, as below

$$\hat{r}_{i,k} = \bigoplus_{\{v | U_i \in \mathcal{C}_v, \mathcal{C}_v \in \mathbf{C}\}} \hat{r}_{i,k,v}. \tag{13}$$

For a hard decision on a singleton $\theta_i \in \Theta$, the pignistic probability function is applied, and then the singleton that has the highest probability is selected to be the preference label. In case a preference label (a singleton or a composite) is needed, the maximum belief with non-overlapping interval strategy [65] is applied; if such as preference label can not be found, the decision is made according to the favor of composite preference label which has the maximum belief and those singletons have a higher plausibility [11].

For the sake of convenience, the proposed system is summarized in two algorithms as shown in Algorithms 1 and 2.

Algorithm 1 Predicting unprovided ratings and computing user-user similarities

```
Input: Rating matrix R, social network G and context information
Output: User-user similarities for each community in G
identify overlapping communities in G via the SLPA algorithm {assume that G consists of V communities}
separate rating matrix R into V sub rating matrix R_1, R_2, ..., R_V according to V communities
for all C_v ∈ 𝓒 do
    for all I_k ∈ I do
        compute group preferences of users, who have rated item I_k, on item I_k via equation (1)
    end for
    for all U_i ∈ C_v do
        for all I_k ∉ ^R I_{i,v} do
            compute concept preferences of user U_i on item I_k via equations (2) and (3)
            compute context preference of user U_i on item I_k via equation (4)
            if context preference of user U_i on item I_k is vacuous then
                assign the value to unprovided rating r_{i,k,v} via equation (6)
            else
                assign the value to unprovided rating r_{i,k,v} via equation (5)
            end if
        end for
    end for
    generate user-user similarity matrix S_v via equations (8), (9) and (10)
end for
```

IV. System Implementation and Discussion

For evaluating recommender systems supporting hard ratings, one can use a number of well-known assessment methods, for example, MAE (Mean Absolute Error), $Precision$, $Recall$, and F_β [66]. Most recently, researchers have developed some new assessment methods which are capable of

Algorithm 2 Generating recommendations

Input: A request to generate recommendations to user U_i
Output: A short list items $\mathcal{L}$

```
for each item I_k which has been not rated by user U_i do
    for each {C_v ∈ C | U_i ∈ C_v} do
        select N_{i,k,v} for user U_i regarding item I_k in community C_v
        estimate rating for user U_i on item I_k in community C_v via equation (11)
    end for
    if U_i belongs to only one community C_v then
        estimate last rating for user U_i on item I_k via equation (12)
    else
        estimate last rating for user U_i on item I_k via equation (13)
    end if
end for
rank items which are not rated by user U_i
generate L
```

measuring performances of recommender systems that use soft ratings, such as $DS\text{-}Precision$, $DS\text{-}Recall$ [62] and $DS\text{-}MAE$, $DS\text{-}F_\beta$ [11]. Let us denote that the last estimated rating (which is used for generating recommendation) of user U_i on item I_k is $\hat{r}_{i,k}$; and the pignistic probability distribution is applied in the mass function $\hat{r}_{i,k}$ is represented as $\widehat{Bp}_{i,k}$. The new assessment methods can be described as below

$$\begin{aligned} DS\text{-}Precision(\theta_j) &= \frac{TP(\theta_j)}{TP(\theta_j)+FP(\theta_j)}; \\ DS\text{-}Recall(\theta_j) &= \frac{TP(\theta_j)}{TP(\theta_j)+FN(\theta_j)}; \\ DS\text{-}MAE(\theta_j) &= \frac{1}{\mid D_j \mid}\sum_{(i,k)\in D_j;\theta_l\in\Theta}\widehat{Bp}_{i,k}(\theta_l)\mid\theta_j-\theta_l\mid; \\ DS\text{-}F_\beta(\theta_j) &= \frac{(\beta^2+1)DS\text{-}Precision(\theta_j)DS\text{-}Recall(\theta_j)}{\beta^2 DS\text{-}Precision(\theta_j)+DS\text{-}Recall(\theta_j)}, \end{aligned}$$

with D_j is the testing set identifying the user-item pairs whose true rating is $\theta_j \in \Theta$, $\beta \geq 1$, and

$$\begin{aligned} TP(\theta_j) &= \sum_{(i,k)\in D_j}\widehat{Bp}_{i,k}(\theta_j); \\ FP(\theta_j) &= \sum_{(i,k)\in D_l;j\neq l}\widehat{Bp}_{i,k}(\theta_j); \\ FN(\theta_j) &= \sum_{(i,k)\in D_j}\widehat{Bp}_{i,k}(\theta_l). \end{aligned}$$

To evaluate the proposed system, we adopted all assessment methods mentioned above. Also, we selected CoFiDS as a baseline for performance comparison.

Since the proposed system requires a domain with soft ratings, we obtained the method suggested in [11] for generating data sets for the experiments. According to this method, data sets containing hard rating are selected first, and then Dempster-Shafer modeling functions are applied to transform hard ratings into corresponding soft ratings.

Regarding the literature, MovieLens data set is widely used for evaluating recommender systems. Additionally, social network information is not available in this data set. Thus, it is suitable for only evaluating the influence of the new method for computing user-user similarities on the performance of recommendations in the proposed system. Moreover, as observed, Flixster data set consists of rating data, friend relations (social network) as well as context information; thus, this data set can be used for fully evaluating the proposed system.

In the experiments, we selected both MovieLens and Flixster data sets. Note that, in these data sets, information about genres in which a user interested is not available; therefore, we assume that the genres in which user $U_i \in \mathbf{U}$ is interested are assigned by genres of items which have been rated by the user.

The prototype application that implements the proposed system was mainly built by using SQL Server 2012 Standard Edition and Visual Basic 6.0. In addition, the experiments were conducted in the environment as follows

- Processor: Intel (R) Core (TM) i5-43000U CPU @1.90 GHz 2.50 GHz
- System type: 64-bit operating system, x64-based processor
- Installed memory (RAM): 4.00 GB
- Operation system: Windows 8.1 Enterprise

In the rest of this section, details of the experiments on the selected data sets will be presented.

A. *Experiment on MovieLens Data Set*

MovieLens data set[1], MovieLens 100k, contains 100,000 hard ratings from 943 users on 1682 movies with a rating domain containing 5 elements, $\Theta = \{1,2,3,4,5\}$. In this data set, each user has rated at least 20 movies, a hard rating $\theta_l \in \Theta$ corresponding to the evaluation of user U_i on Item I_k is transformed into a soft rating $r_{i,k}$ by the Dempster-Shafer modeling function suggested in [11], as shown below

$$r_{i,k}(A) = \begin{cases} \alpha_{i,k}(1-\sigma_{i,k}), & \text{for } A=\{\theta_l\}; \\ \alpha_{i,k}\sigma_{i,k}, & \text{for } A=B; \\ 1-\alpha_{i,k}, & \text{for } A=\Theta; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{with } B = \begin{cases} \{\theta_1,\theta_2\}, & \text{if } l=1; \\ \{\theta_{L-1},\theta_L\}, & \text{if } l=L; \\ \{\theta_{l-1},\theta_l,\theta_{l+1}\}, & \text{otherwise.} \end{cases}$$

Here, $\alpha_{i,k} \in [0,1]$ and $\sigma_{i,k} \in [0,1]$ are a trust factor and a dispersion factor, respectively [11].

[1] http://grouplens.org/datasets/movielens/

TABLE II: Overall MAE versus w_1 and w_2 (MovieLens)

w_2 \ w_1	0.0	0.1	0.2	0.3	0.4	0.5
0.0	0.8387	0.8381	0.8377	0.8377	0.8362	0.8360
0.1	0.8385	0.8381	0.8379	0.8379	0.8362	
0.2	0.8387	0.8381	0.8379	0.8379	0.8362	
0.3	0.8362	0.8362	0.8362	0.8362		
0.4	0.8385	0.8383	0.8377	0.8377		
0.5	0.8385	0.8383	0.8375			
0.6	0.8383	0.8383	0.8373			
0.7	0.8381	0.8381				
0.8	0.8379	0.8377				
0.9	0.8377					
1.0	0.8377					

TABLE III: Overall $DS\text{-}MAE$ versus w_1 and w_2 (MovieLens)

w_2 \ w_1	0.0	0.1	0.2	0.3	0.4	0.5
0.0	0.8359	0.8352	0.8346	0.8346	0.8338	0.8335
0.1	0.8357	0.8351	0.8347	0.8347	0.8339	
0.2	0.8358	0.8351	0.8346	0.8346	0.8339	
0.3	0.8339	0.8339	0.8339	0.8339		
0.4	0.8356	0.8351	0.8346	0.8346		
0.5	0.8355	0.8351	0.8345			
0.6	0.8354	0.8351	0.8345			
0.7	0.8354	0.8350				
0.8	0.8353	0.8349				
0.9	0.8351					
1.0	0.8351					

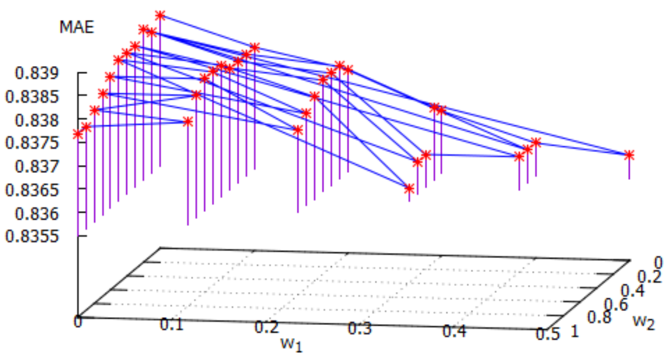

Fig. 3: Visualizing overall MAE (MovieLens)

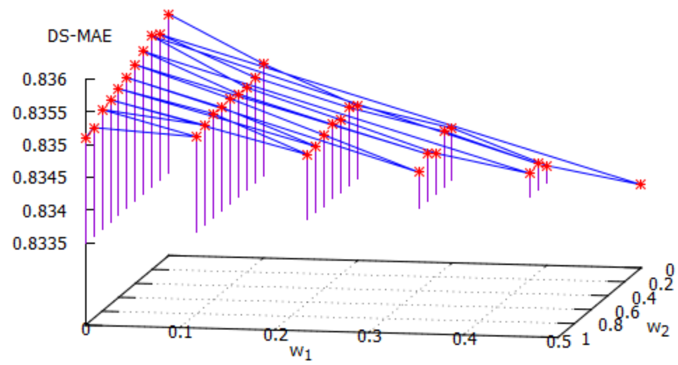

Fig. 4: Visualizing overall $DS\text{-}MAE$ (MovieLens)

The context information in MovieLens data set, which is considered for grouping user, is represented as follows

$$\begin{aligned} \mathbf{C} &= \{C_1\} = \{Genre\}; \\ C_1 &= \{G_{1,1}, G_{1,2}, ..., G_{1,19}\} \\ &= \{Unknown, Adventure, Action, Animation, \\ &Children's, Comedy, Drama, Documentary, Crime, \\ &Musical, Film\text{-}Noir, Fantasy, Horror, Western, \\ &Sci\text{-}Fi, Romance, Thriller, War, Mystery\}. \end{aligned}$$

The unprovided ratings were generated by using equations (1), (2), (3), (4), (5) and (6). Note that when generating unprovided ratings by using the method introduced in [11] (applying equations (1), (2), (3), (4) and (6)), some unprovided ratings are still vacuous, as illustrated in Example 1.

Example 1. In MovieLens data set, let us consider user U_c with $f_1(U_c) = \{G_{1,4}, G_{1,5}, G_{1,6}, G_{1,18}\} = \{Animation, Comedy, Children's, War\}$ and item I_t with $g_1(I_t) = \{G_{1,17}\} = \{Thriller\}$. Supposing that user U_c has not rated item I_t; and we need to generate unprovided rating $r_{c,t}$. According the method in introduced in [11], the predicting process is as below

- Regarding equation (1), we have

$$^{G}m_{1,17,t} = \bigoplus_{\{j \mid I_t \in {}^{R}\mathbf{U}_j, G_{1,17} \in f_1(U_j)\}} r_{j,t};$$
$$^{G}m_{1,q,t} = vacuous, \forall G_{1,q} \in C_1 \text{ and } q \neq 17.$$

- Using equations (2) and (3), we obtain

$$^{C}m_{c,1,t} = \bigoplus_{\{q \mid G_{1,q} \in f_1(U_c) \cap g_1(I_t)\}} {}^{G}m_{1,q,t} = vacuous.$$

- According to equation (4), we get

$$^{\mathbf{C}}m_{c,t} = {}^{C}m_{c,1,t} = vacuous.$$

- Applying equation (6), we have

$$r_{c,t} = {}^{\mathbf{C}}m_{c,t} = vacuous.$$

For being consistent with the baseline, we employed the method introduced in [64] to generate the tesing as well as training data. First, we randomly selected 10% of users in MovieLens data set; and then, for each selected user, we randomly withheld 5 ratings. Next, we used the withheld and remaining ratings as testing as well as training data, respectively. We repeated this process for 10 times in order to generate 10 different splits. The average results obtained from 10 independent experiments on 10 splits were represented in this section. Also, in the experiments, we selected values for some parameters as follows: $\beta = 1, \gamma = 10^{-5}$, $\forall(i,k)\{\alpha_{i,k}, \sigma_{i,k}\} = \{0.9, 2/9\}$.

To evaluate the impacts of two reliability coefficients of the new method for computing user-user similarities, we chose $K = 18$ and $\tau = 0$, and then measured the performance of recommendations. The performance according to evaluation criteria MAE and $DS\text{-}MAE$ are presented in Tables II and III, respectively. As we can see, reliability coefficent w_1 almost linearly influences the performance, and reliability coefficient w_2 has a slight effect. As observed, in MovieLens data set, for any two users, the number of movies rated by both of them is very small or equal to zero; that is why the impact of reliability coefficient w_2 is very weak. The information in Tables II and III is visualized in Fig. 3 and Fig. 4, respectively. Note that, the finding is almost the same with other evaluation criteria.

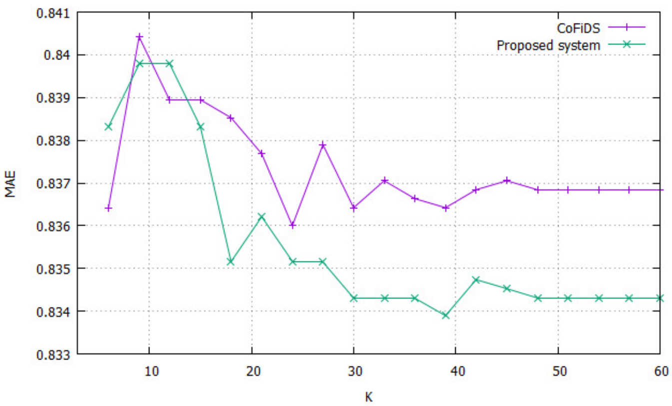

Fig. 5: Overall MAE versus K (MovieLens)

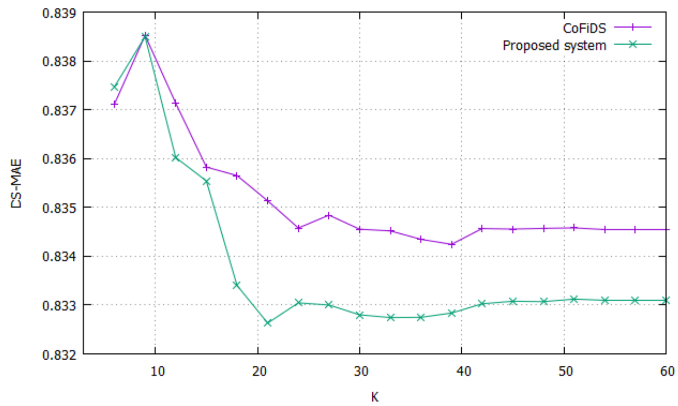

Fig. 6: Overall $DS\text{-}MAE$ versus K (MovieLens)

TABLE IV: The comparison in hard decisions (MovieLens)

Metric	True Rating					Overall
	1	2	3	4	5	
Proposed system:						
$Precision$	0.1770	0.2242	**0.3206**	**0.3919**	**0.4484**	**0.3641**
$Recall$	0.0152	0.0924	**0.3158**	**0.6642**	0.1851	**0.3718**
MAE	**2.4075**	1.5087	**0.7382**	**0.3690**	1.0157	**0.8343**
F_1	0.0649	0.1434	**0.3175**	**0.4923**	**0.2592**	**0.3468**
CoFiDS:						
$Precision$	0.1770	**0.2253**	0.3177	0.3903	0.4375	0.3600
$Recall$	0.0152	0.0924	0.3140	0.6583	0.1851	0.3693
MAE	2.4117	**1.5046**	0.7426	0.3748	**1.0149**	0.8371
F_1	0.0649	**0.1436**	0.3151	0.4894	0.2573	0.3455

TABLE V: The comparison in soft decisions (MovieLens)

DS -Metric	True Rating					Overall
	1	2	3	4	5	
Proposed system:						
$Precision$	**0.1749**	**0.2300**	**0.3175**	**0.3908**	**0.4462**	**0.3609**
$Recall$	0.0156	**0.0949**	0.3164	**0.6605**	0.1815	**0.3702**
MAE	**2.4066**	**1.4918**	**0.7344**	**0.3713**	1.0175	**0.8327**
F_1	0.0267	**0.1329**	**0.3161**	**0.4903**	0.2553	**0.3315**
CoFiDS:						
$Precision$	0.1709	0.2294	0.3172	0.3903	0.4405	0.3589
$Recall$	**0.0158**	0.0942	0.3164	0.6572	**0.1831**	0.3694
MAE	2.4088	1.4944	0.7363	0.3749	**1.0161**	0.8345
F_1	**0.0271**	0.1322	0.3160	0.4890	**0.2560**	0.3311

To compare with CoFiDS, selected $w_1 = 0.5, w_2 = 0, \tau = 0$, and K ranging from 6 to 60 with step size 3. The results of comparison according to evaluation criteria MAE and $DS\text{-}MAE$ are shown in Fig. 5 and Fig. 6, respectively. As observed Fig. 5, the performances of the proposed system as well as CoFiDS are fluctuated with $K < 42$, and appear to be stable when $K \geq 42$; the reason is that, in the training data, with $K \geq 42$, the number of members in neighborhood sets becomes stable. Especially, both figures show that, with $K \geq 15$, the proposed system is better than the baseline; it means that weighting predicted ratings weaker than provided ones helps to improve the quality of recommendations.

Tables IV and V depict the results of comparison between the proposed system and the baseline in hard and soft decisions with $w_1 = 0.5, w_2 = 0, \tau_2 = 0$, and $K = 33$. In these tables, each rating value is presented in a column, and the bold and underlined values indicate the better and the equal performances according to each category, respectively. As detailed in two tables, the proposed system is more effective than the baseline in all selected measurement criteria. Because MovieLens data set consists of a small number of provided ratings, the absolute values of the performance of the proposed system are just slightly higher than those of CoFiDS. In fact, if more provided ratings are available in the data set, the proposed system the proposed system could be much better than the baseline.

B. Experiment on Flixster Data Set

Flixster data set[2] consists of friend relationships and hard ratings with rating values ranging from 0.5 to 5 with step size 0.5. Notably, we have enriched this data set by crawling the genres of movies. After crawling and cleaning, we achieved a new Flixster data set that contains 49,410 friend relationships, 535,013 hard ratings from 3,827 users on 1210 movies.

In addition, in the new data set, each user has rated at least 15 movies, and the genres that is considered as context information are represented as follows

$$\begin{aligned} \mathbf{C} &= \{Genre\}; \\ Genre &= \{Drama, Comedy, Action \ \& \ Adventure, \\ &Television, Mystery \ \& \ Suspense, Horror, \\ &Science\ Fiction \ \& \ Fantasy, Kids \ \& \ Family, \\ &Art\ House \ \& \ International, Romance, Classics, \\ &Musical \ \& \ Performing\ Arts, Anime \ \& \ Manga, \\ &Animation, Western, Documentary, Special\ Interest, \\ &Sports \ \& \ Fitness, Cult\ Movies\}. \end{aligned}$$

To transform a hard rating $\theta_l \in \Theta$ according the evaluation of user U_i on item I_k into a soft rating $r_{i,k}$, we applied the

[2]https://www.cs.ubc.ca/ jamalim/datasets/

TABLE VI: Overall MAE versus w_1 and w_2 (Flixster)

		w_1					
		0.0	0.1	0.2	0.3	0.4	0.5
	0.0	0.8337	0.8337	0.8338	0.8333	0.8338	0.8338
	0.1	0.8337	0.8335	0.8338	0.8334	0.8336	
	0.2	0.8339	0.8335	0.8337	0.8336	0.8337	
	0.3	0.8339	0.8337	0.8338	0.8334		
	0.4	0.8339	0.8339	0.8337	0.8338		
w_2	0.5	0.8339	0.8339	0.8336			
	0.6	0.8339	0.8338	0.8337			
	0.7	0.8337	0.8337				
	0.8	0.8337	0.8337				
	0.9	0.8337					
	1.0	0.8337					

TABLE VII: Overall $DS\text{-}MAE$ versus w_1 and w_2 (Flixster)

		w_1					
		0.0	0.1	0.2	0.3	0.4	0.5
	0.0	0.8340	0.8337	0.8338	0.8338	0.8340	0.8337
	0.1	0.8340	0.8338	0.8337	0.8339	0.8339	
	0.2	0.8341	0.8337	0.8337	0.8340	0.8339	
	0.3	0.8339	0.8338	0.8337	0.8339		
	0.4	0.8339	0.8338	0.8338	0.8340		
w_2	0.5	0.8340	0.8339	0.8338			
	0.6	0.8341	0.8339	0.8339			
	0.7	0.8339	0.8339				
	0.8	0.8339	0.8339				
	0.9	0.8339					
	1.0	0.8339					

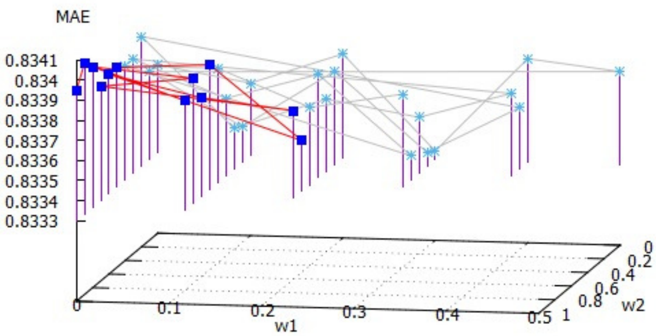

Fig. 7: Visualizing overall MAE (Flixster)

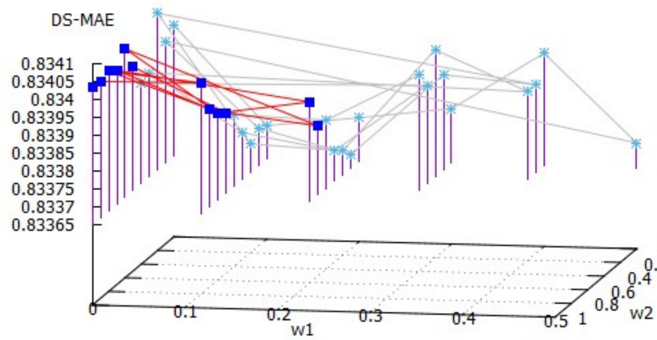

Fig. 8: Visualizing overall $DS\text{-}MAE$ (Flixster)

Dempster-Shafer modeling function which is defined below

$$r_{i,k}(A) = \begin{cases} \alpha_{i,k}(1-\sigma_{i,k}), & \text{for } A = \{\theta_l\}; \\ \frac{3}{5}\alpha_{i,k}\sigma_{i,k}, & \text{for } A = B; \\ \frac{2}{5}\alpha_{i,k}\sigma_{i,k}, & \text{for } A = C; \\ 1-\alpha_{i,k}, & \text{for } A = \Theta; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{where } B = \begin{cases} \{\theta_1, \theta_2\}, & \text{if } l = 1; \\ \{\theta_{L-1}, \theta_L\}, & \text{if } l = L; \\ \{\theta_{l-1}, \theta_l, \theta_{l+1}\}, & \text{otherwise,} \end{cases}$$

$$\text{and } C = \begin{cases} \{\theta_1, \theta_2, \theta_3\}, & \text{if } l = 1; \\ \{\theta_1, \theta_2, \theta_3, \theta_4\}, & \text{if } l = 2; \\ \{\theta_{L-3}, \theta_{L-2}, \theta_{L-1}, \theta_L\}, & \text{if } l = L-1; \\ \{\theta_{L-2}, \theta_{L-1}, \theta_L\}, & \text{if } l = L; \\ \{\theta_{l-2}, \theta_{l-1}, \theta_l, \theta_{l+1}, \theta_{l+2}\}, & \text{otherwise,} \end{cases}$$

with $\alpha_{i,k} \in [0,1]$ and $\sigma_{i,k} \in [0,1]$ are is a trust factor and a dispersion factor, respectively [11].

In Flixster data set, all users belong to a social network whose nodes are connected by undirected friendships. To detect communities in this network, we selected $\mathtt{M}_{min} = 100$, $\mathtt{M}_{max} = 500$, and $T = 100$ for SLPA algorithm. After executing this algorithm, we obtained 7 overlapping communities (The number of members in each communities is as following: 226, 377, 2749, 712, 1011, 460, and 105).

For each user in Flixster data set, we randomly withheld 5 ratings. The withheld ratings were employed as the testing data, and the remaining ratings were considered to be the training data. Moreover, in the experiments, we chose values for some other parameters as follows: $\beta = 1, \gamma = 10^{-5}$, and $\forall(i,k)\{\alpha_{i,k}, \sigma_{i,k}\} = \{0.9, 2/9\}$.

To measure the impact of coefficients w_1 and w_2, we selected $K = 25$ and $\tau = 0.75$ for the experiments. Tables VI and VII represent results of overall MAE and $DS\text{-}MAE$ criteria, respectively. It can be seen in these tables, when $w_1 \leq 0.2$ and $w_2 \geq 0.5$, the performance of the proposed system is mostly linearly dependent on the coefficients; in the other cases, only some values of the coefficients effect the proposed system. The information presented in Table VI and Table VII is visualized in Fig. 7 and Fig. 8, respectively.

To compare with CoFiDS, we selected $w_1 = 0.2, w_2 = 0.5, \tau = 0$, and K ranging from 5 to 150 with step size 5 for the experiments. The results of comparison are shown in Fig. 9 and Fig. 10. According to these figures, the proposed system is more effective than the baseline in both hard and soft decisions in all cases, especially when $K \leq 45$. These results indicate that using community context information for predicting unprovided ratings and assigning weights to rating data when computing user-user similarity is capable of improving the performance of recommendations.

Tables VIII and IX show the results of comparison between the proposed system and the baseline in both hard as well as soft decisions with $K = 40$ and $\tau = 0$. In these tables, each rating value is presented in a column; bold values illustrate the better performances, underlined values indicate equal performances, and italic values mention that they are incomparable for comparison. In addition, the columns regarding rating values ranging from 1.0 to 2.5 contain some values as 0 or N/A (Not applicable) because, in Flixster data set, the number

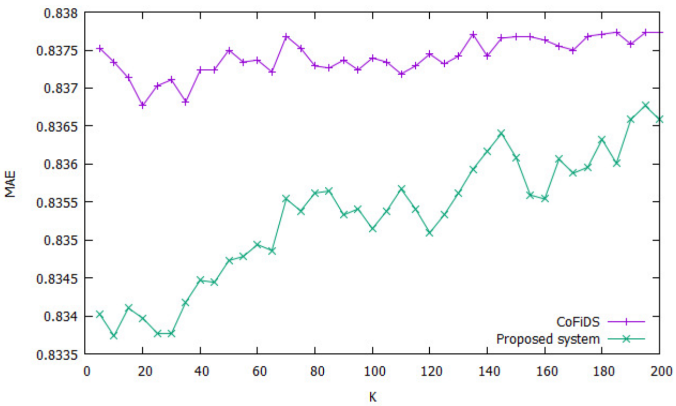

Fig. 9: Overall MAE versus K (Flixster)

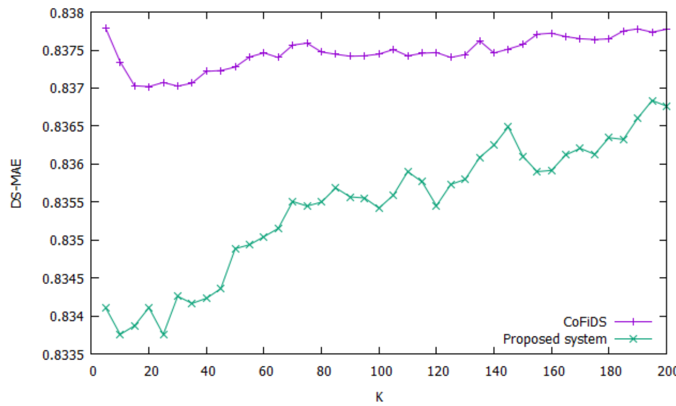

Fig. 10: Overall $DS\text{-}MAE$ versus K (Flixster)

TABLE VIII: The comparison in hard decisions (Flixster)

$Metric$	True rating value										Overall
	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	
Proposed system:											
MAE	**3.3170**	**2.8783**	**2.3949**	**1.8790**	**1.3992**	**0.8983**	**0.4795**	0.1515	0.5722	**0.9740**	**0.8346**
$Precision$	0.8750	*0*	*0*	*0*	*0.2000*	**0.2100**	**0.1789**	**0.2019**	**0.1616**	0.3811	*0.2357*
$Recall$	0.0221	0	0	0	**0.0010**	**0.0674**	**0.1477**	0.7809	**0.0138**	**0.0911**	**0.2099**
F_1	0.0431	*N/A*	*N/A*	*N/A*	*0.0020*	0.**1021**	**0.1618**	0.3208	**0.0254**	**0.1470**	*N/A*
CoFiDS:											
MAE	3.3281	2.9006	2.4205	1.898	1.4052	0.9017	0.4796	**0.1226**	**0.5701**	0.998	0.8372
$Precision$	0.8750	*N/A*	*N/A*	*N/A*	*N/A*	0.1998	0.1778	0.2005	0.1000	**0.3960**	*N/A*
$Recall$	0.0221	0	0	0	0	0.0607	0.122	**0.8247**	0.0013	0.07	0.2069
F_1	0.0431	*N/A*	*N/A*	*N/A*	*N/A*	0.0931	0.1447	**0.3226**	0.0026	0.1189	*N/A*

TABLE IX: The comparison in soft decisions (Flixster)

$DS\text{-}Metric$	True rating value										Overall
	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	
Proposed system:											
MAE	**3.3172**	**2.8811**	**2.3965**	**1.879**	**1.3969**	**0.8994**	**0.4794**	0.1519	0.5710	**0.9725**	**0.8342**
$Precision$	**0.8637**	0	0	**0.1406**	**0.1152**	**0.2088**	**0.1787**	**0.2018**	**0.1682**	0.3813	**0.2368**
$Recall$	0.0221	0	0	**0.0007**	**0.0005**	**0.0667**	**0.1478**	0.7796	**0.0148**	**0.0920**	**0.2100**
F_1	0.0431	0	0	**0.0015**	**0.0011**	**0.1011**	**0.1618**	0.3206	**0.0273**	**0.1482**	**0.1426**
CoFiDS:											
MAE	3.3283	2.9014	2.4208	1.8971	1.4060	0.9009	0.4802	**0.1226**	**0.5706**	0.9976	0.8372
$Precision$	0.8750	0	0	0.0001	0.0002	0.2031	0.1770	0.2006	0.0966	**0.3965**	0.2198
$Recall$	0.0221	0	0	0	0	0.0617	0.1217	**0.8245**	0.0014	0.0702	0.207
F_1	0.0431	0	0	0	0	0.0947	0.1442	**0.3227**	0.0028	0.1193	0.1292

of users who have rated as 1.0, 1.5, 2.0 or 2.5 is very small compared to the number of other users who have rated as higher values.

As detailed in Tables VIII and IX, the proposed system is more effective than the baseline in most of true rating values. However, similar to the results obtained from the experiments on MovieLens data set, the absolute values of the performance of the proposed system are just slightly higher than those of the baseline. In fact, the different absolute values could be much greater if the data set contains more provided ratings and overlapping communities in the social network are detected by using another information such as the frequency of communication, the number of comments, the common interests, and so on.

V. Conclusion

In summary, in this paper, we have developed a new collaborative filtering recommender system that employs DST for representing ratings, and uses community context information for predicting unprovided ratings. Suitable recommendations to users are generated by using both predicted and provided ratings with the important aspect being the stipulation that provided ratings are more significant than predicted ones. Remarkably, the developed system is capable of dealing with not only imperfect information about user preferences but also the sparsity problem. Moreover, the experiment results show that performance of the new system has improved in both hard and soft decisions compared with a similar system, CoFiDS.

ACKNOWLEDGMENT

The authors are grateful to the anonymous referees for their constructive comments as well as suggestions. This research work was partly supported by JSPS KAKENHI Grant No. 25240049.

REFERENCES

[1] V.-D. Nguyen and V.-N. Huynh, "A community-based collaborative filtering system dealing with sparsity problem and data imperfections," in *PRICAI*, 2014, pp. 884–890.

[2] ——, "A reliably weighted collaborative system." in *ECSQARU*, 2015, pp. 429–439.

[3] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005.

[4] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, 2013.

[5] S. Jiang, X. Qian, T. Mei, and Y. Fu, "Personalized travel sequence recommendation on multi-source big social media," *IEEE Trans. Big Data*, vol. 2, no. 1, pp. 43–56, 2016.

[6] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *WWW*, 2001, pp. 285–295.

[7] D. Rafailidis and A. Nanopoulos, "Modeling users preference dynamics and side information in recommender systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 6, pp. 782–792, 2016.

[8] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, 2003.

[9] J. B. Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation applications," *Data Min. Knowl. Discov.*, vol. 5, no. 1, pp. 115–153, 2001.

[10] Q. Shambour and J. Lu, "A hybrid trust-enhanced collaborative filtering recommendation approach for personalized government-to-business e-services," *Int. J. Intell. Syst.*, vol. 26, no. 9, pp. 814–843, 2011.

[11] T. Wickramarathne, K. Premaratne, M. Kubat, and D. Jayaweera, "CoFiDS: A belief-theoretic approach for automated collaborative filtering," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 2, pp. 175–189, 2011.

[12] S. Y. Jung, J. Hong, and T. Kim, "A statistical model for user preference," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 834–843, 2005.

[13] W. Feller, *An introduction to probability theory and its applications*. J. Wiley & Sons, 1968, vol. 1.

[14] A. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *Ann. Math. Stat.*, vol. 38, pp. 325–339, 1967.

[15] G. Shafer, *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[16] L. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Set. Syst.*, vol. 1, pp. 3–28, 1978.

[17] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., *Recommender Systems Handbook*. Springer, 2011.

[18] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 116–142, 2004.

[19] L. Tang and H. Liu, *Community detection and mining in social Media*, ser. Synthesis Lectures on Data Mining and Knowledge Discovery. Morgan & Claypool Publishers, 2010.

[20] P. Smets and R. Kennes, "The transferable belief model," *Artif. Intell.*, vol. 66, no. 2, pp. 191–234, 1994.

[21] P. Smets, "Practical uses of belief functions," in *UAI*, 1999, pp. 612–621.

[22] V.-N. Huynh, "Discounting and combination scheme in evidence theory for dealing with conflict in information fusion," in *MDAI*, 2009, pp. 217–230.

[23] J. Xie and B. Szymanski, "Towards linear time overlapping community detection in social networks," in *PAKDD (2)*, 2012, pp. 25–36.

[24] U. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Phys. Rev. E*, vol. 76, no. 3, pp. 036 106+, 2007.

[25] M. Girvan and M. Newman, "Community structure in social and biological networks," *Proc. Natl. Acad. Sci.*, vol. 99, no. 12, pp. 7821–7826, 2002.

[26] S. Gregory, "An algorithm to find overlapping community structure in networks," in *PKDD*, 2007, pp. 91–102.

[27] ——, "A fast algorithm to find overlapping communities in networks," in *ECML/PKDD (1)*, 2008, pp. 408–423.

[28] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[29] M. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, 2003.

[30] ——, "Modularity and community structure in networks," *Proc. Natl. Acad. Sci.*, vol. 103, no. 23, pp. 8577–8582, 2006.

[31] J. Xie, S. Kelley, and B. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, pp. 43:1–43:35, 2013.

[32] S. Jiang, X. Qian, J. Shen, Y. Fu, and T. Mei, "Author topic model-based collaborative filtering for personalized POI recommendations," *IEEE Trans. Multimedia*, vol. 17, no. 6, pp. 907–918, 2015.

[33] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Comput.*, vol. 42, no. 8, pp. 30–37, 2009.

[34] X. Qian, H. Feng, G. Zhao, and T. Mei, "Personalized recommendation combining user interest and social circle," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 7, pp. 1763–1777, 2014.

[35] L. Pu and B. Faltings, "Understanding and improving relational matrix factorization in recommender systems," in *RecSys*, 2013, pp. 41–48.

[36] J. Sun, D. Parthasarathy, and K. Varshney, "Collaborative kalman filtering for dynamic matrix factorization," *IEEE Trans. Sign. Proc.*, vol. 62, no. 14, pp. 3499–3509, 2014.

[37] A. Hernando, J. Bobadilla, and F. Ortega, "A non negative matrix factorization for collaborative filtering recommender systems based on a bayesian probabilistic model," *Knowl.-Based Syst.*, vol. 97, pp. 188–202, 2016.

[38] C. Jiang, R. Duan, H. K. Jain, S. Liu, and K. Liang, "Hybrid collaborative filtering for high-involvement products: A solution to opinion sparsity and dynamics," *Decis. Support Syst.*, vol. 79, pp. 195–208, 2015.

[39] H. Kim, A. Ji, I. Ha, and G. Jo, "Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation," *Electron. Commer. Res. Appl.*, vol. 9, no. 1, pp. 73–83, 2010.

[40] M. Grcar, D. Mladenic, B. Fortuna, and M. Grobelnik, "Data sparsity issues in the collaborative filtering framework," in *WebKDD*, 2005, pp. 58–76.

[41] G. Zhao, X. Qian, X. Lei, and T. Mei, "Service quality evaluation by exploring social users' contextual information," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 12, pp. 3382–3394, 2016.

[42] M. Eirinaki, M. D. Louta, and I. Varlamis, "A trust-aware system for personalized user recommendations in social networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 4, pp. 409–421, 2014.

[43] X. Lei, X. Qian, and G. Zhao, "Rating prediction based on social sentiment from textual reviews," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1910–1921, 2016.

[44] X. Yang, H. Steck, and Y. Liu, "Circle-based recommendation in online social networks," in *KDD*, 2012, pp. 1267–1275.

[45] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang, "Social contextual recommendation," in *CIKM*, 2012, pp. 45–54.

[46] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation," in *SIGIR*, 2009, pp. 195–202.

[47] S. M. Kywe, E. Lim, and F. Zhu, "A survey of recommender systems in Twitter," in *SocInfo*, 2012, pp. 420–433.

[48] C. D. Maio, G. Fenza, V. Loia, and M. Parente, "Time aware knowledge extraction for microblog summarization on Twitter," *Inf. Fusion*, vol. 28, pp. 60–74, 2016.

[49] Z. Sun, L. Han, W. Huang, X. Wang, X. Zeng, M. Wang, and H. Yan, "Recommender systems based on social networks," *J. Syst. Software*, vol. 99, pp. 109–119, 2015.

[50] G. Zhao, X. Qian, and X. Xie, "User-service rating prediction by exploring social users' rating behaviors," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 496–506, 2016.

[51] M. Jiang, P. Cui, F. Wang, W. Zhu, and S. Yang, "Scalable recommendation with social contextual information," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 11, pp. 2789–2802, 2014.

[52] G. Guo, J. Zhang, and D. Thalmann, "Merging trust in collaborative filtering to alleviate data sparsity and cold start," *Knowl.-Based Syst.*, vol. 57, pp. 57–68, 2014.

[53] H. Wu, K. Yue, Y. Pei, B. Li, Y. Zhao, and F. Dong, “Collaborative topic regression with social trust ensemble for recommendation in social media systems,” *Knowl.-Based Syst.*, vol. 97, pp. 111–122, 2016.

[54] Y. Kim and K. Shim, “TWILITE: A recommendation system for Twitter using a probabilistic model based on latent Dirichlet allocation,” *Inf. Syst.*, vol. 42, pp. 59–77, 2014.

[55] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, “Recommender systems with social regularization,” in *WSDM*, 2011, pp. 287–296.

[56] V.-N. Huynh, J. Lawry, and Y. Nakamori, “Integrated uncertainty management for decision making,” *Ann. Oper. Res.*, vol. 195, no. 1, pp. 1–4, 2012.

[57] V.-N. Huynh, Y. Nakamori, T. B. Ho, and T. Murai, “Multiple-attribute decision making under uncertainty: the evidential reasoning approach revisited,” *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 4, pp. 804–822, 2006.

[58] L. Jiao, T. Denoeux, and Q. Pan, “A hybrid belief rule-based classification system based on uncertain training data and expert knowledge,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 12, pp. 1711–1723, 2016.

[59] V.-D. Nguyen and V.-N. Huynh, “Two-probabilities focused combination in recommender systems,” *Int. J. Approx. Reasoning*, vol. 80, pp. 225–238, 2017.

[60] Q. Luo, Y. Peng, X. Peng, and A. El-Saddik, “Uncertain data clustering-based distance estimation in wireless sensor networks,” *Sensors*, vol. 14, no. 4, pp. 6584–6605, 2014.

[61] V.-D. Nguyen and V.-N. Huynh, “Integrating with social network to enhance recommender system based-on Dempster-Shafer theory,” in *CSoNet*, 2016, pp. 170–181.

[62] K. Hewawasam, K. Premaratne, and M.-L. Shyu, “Rule mining and classification in a situation assessment application: A belief-theoretic approach for handling data imperfections,” *IEEE Trans. Syst. Man Cybern., Part B*, vol. 37, no. 6, pp. 1446–1459, 2007.

[63] H. Chan and A. Darwiche, “A distance measure for bounding probabilistic belief change,” *Int. J. Approx. Reasoning*, vol. 38, no. 2, pp. 149–174, 2005.

[64] J. Herlocker, J. Konstan, A. Borchers, and J. Riedl, “An algorithmic framework for performing collaborative filtering,” in *SIGIR*, 1999, pp. 230–237.

[65] I. Bloch, “Some aspects of Dempster-Shafer evidence theory for classification of multi-modality medical images taking partial volume effect into account,” *Pattern Recogn. Lett.*, vol. 17, no. 8, pp. 905–919, 1996.

[66] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl, “Evaluating collaborative filtering recommender systems,” *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.

Van-Doan Nguyen received his bachelor’s and master’s degrees in Information Technology from Ho Chi Minh City University of Technology (HCMUT) in 2002 and 2007, respectively. After getting his master’s degree, he became lecturer at HCMUT. In 2013, he won a Japanese Government Scholarship (MEXT Scholarship), and moved to Japan to study at Japan Advanced Institute of Science and Technology (JAIST). He earned his Ph.D. degree in Knowledge Science in March 2017. Currently, he is a postdoctoral research fellow at JAIST. His research interests include recommender systems, information fusion, belief functions, and data analytics.

Van-Nam Huynh is an associate professor at Japan Advanced Institute of Science and Technology, Japan. He received a PhD in Mathematics (1999) from the Institute of Information Technology, Vietnam Academy of Science and Technology, and a “Habilitation à Diriger des Recherches” (2012) at Université de Technologie de Compiègne, France. He was a post-doctoral fellow (2001–2002) awarded by Inoue Foundation for Science at Japan Advanced Institute of Science and Technology (JAIST). His research interests include decision theories, data mining and machine learning, computing and reasoning with words, information fusion, kansei information processing and application. Currently he is a member of the Editorial Board of the International Journal of Approximate Reasoning and an associate editor of the International Journal of Knowledge and Systems Science.

Songsak SRIBOONCHITTA received Bachelor’s and Master Degrees in Economics from Thammasat University, Thailand in 1972 and 1975 respectively and received Ph.D in Agricultural Economics from The University of Illinois at Urbana-Champaign in 1983. He is a professor of economics and has been working at the Faculty of Economics, Chiang Mai University since 1976. He was the Dean of the Faculty of Economics, Chiang Mai University during 2004-2008. Presently he is the President of Thailand Econometric Society and also Director of Centre of Excellence in Econometrics, Chiang Mai University. He has published one book and five edited books and more than 100 papers. His current research interests include econometrics, data mining and decision analysis.