World Wide Web (2015) 18:139–157
DOI 10.1007/s11280-013-0243-3

Trust-aware media recommendation in heterogeneous social networks

Jian Wu · Liang Chen · Qi Yu · Panpan Han · Zhaohui Wu

Received: 30 January 2013 / Revised: 15 June 2013 /
Accepted: 8 July 2013 / Published online: 27 July 2013

Abstract Social network sites, such as Facebook and Twitter, are gaining increasing popularity nowadays by providing a convenient platform for sharing and consuming information of all kinds. While the ever increasing information sources on the social network sites hold tremendous promise, how to select user interested information becomes nontrivial as users are easily overloaded by a vast amount candidate information sources. Furthermore, as the information providers are autonomous entities in an open social network environment, they may spread information that is unreliable or completely fake. Hence, technological advances are in demand to recommend information sources to social network users that both match their interests and come from reliable information sources. We develop a novel social media recommendation framework, referred to as GCCR, to tackle the above central challenges. GCCR is coined based on the key technologies that supports the proposed framework: Graph summarization, Content-based approach, Clustering, and Recommendation. A user-centric strategy is adopted that exploits the historical behavior of a set of *seed* users as evidence to assess the trustworthiness of different information providers. A two-phase process that employs graph summarization

J. Wu · L. Chen (✉) · P. Han · Z. Wu
Computer Science & Technology College, Zhejiang University, Zhejiang, China
e-mail: cliang@zju.edu.cn

J. Wu
e-mail: wujian2000@zju.edu.cn

P. Han
e-mail: ronson@zju.edu.cn

Z. Wu
e-mail: wzh@zju.edu.cn

Q. Yu
College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, USA
e-mail: qi.yu@rit.edu

and content-based clustering is developed to partition users into different interest groups. The interest group information is then used for recommendation purpose. We perform extensive experiments on real-world social network data to assess the effectiveness of the proposed GCCR framework.

Keywords Media recommendation · Trust · Heterogeneous social network

1 Introduction

Social Network Sites (SNSs) (e.g., Twitter [29] and Facebook [7] in US as well as Sina Weibo [27] and Douban [6] in China) are increasingly attracting people's attention nowadays. Different from SNSs such as Facebook and MySpace [21], which are based on two-way friendship relationships, Microblogging sites such as Twitter, do not have to be reciprocal. This implies that there is no need to follow someone back who is following you [17]. Such an unrestricted following mechanism enables a fast growth of Microblogging sites, which facilities large-scale information propagation. It has been noticed that most *follower-followed* relationships on Microblogging sites mimic the traditional subscription relationship between media information subscribers and distributors, in which the subscribers consume media information but have little interaction with the distributors. The one-way subscription based Microblogging sites are in essence heterogeneous. One can classify the nodes of such heterogeneous social networks into two categories: *user nodes*, which represent ordinal users, and *media nodes*, which represent publishing media and other news sources, such as *China Railway*, *Zhejiang University*, and *Hot videos*.

A user node may follow multiple media nodes based on the user's interests. Different user nodes may also follow each other due to their social relationships and common interests. In contrast, media nodes are media publishers and may have many followers but rarely follow other nodes. Figure 1 shows an example of a two-way relationship based social network structure. Figure 2 visualizes the data we extracted from a real-world Microblogging site, Sina Weibo, where the blue nodes represent user nodes and red nodes represent media nodes. The dotted lines represent one-way subscription relationship and solid lines represent two-way friendship relationship.

The large number of social media providers and the consumers that are expected to heavily take advantage of the social media platform to deliver and consume information of interest has led to an exponential growth in both the content and users of SNSs. The ever increasing information sources on SNSs hold tremendous promise by providing a large-scale online repository that hosts information of all kinds. Despite being offered plenty of options for choosing information that fits their interests, users

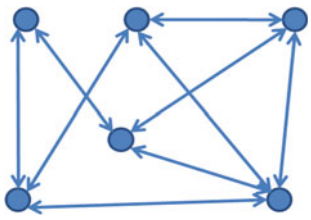


Figure 1 Structure of a two-way relationship based social network

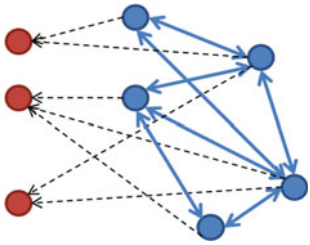


Figure 2 Structure of a heterogeneous social network extracted from Sina Weibo

can be easily overwhelmed by a vast amount candidate information sources. Things may become more complicated as information providers are autonomous entities in an open social network environment and they may deliver information that is unreliable or completely fake. However, there currently lacks a systematic support to recommend information sources to social network users that both match their interest and come from reliable sources.

In this paper, we aim to develop a trust-aware social media recommendation framework, referred to as GCCR, to tackle the key challenges as highlighted above. Traditional security based technologies, such as authentication, authorization, and encryption, ensure a secure interaction between a user and a media service provider. Nonetheless, they do not offer much control with respect to the delivered information. Nor can the social network users rely on the security mechanisms to select the trustworthy media providers that can provide reliable information that matches their personal interests. Both research findings (e.g., [5]) and real-world systems (e.g., Ebay and Amazon) have suggested that the reputation based approach can help improve user's trust in an open and distributed environment. Reputation systems usually use a feedback mechanism where the feedback values (or ratings) reflect the perception of other users on a given service provider through previous transactions with this provider [20, 30]. Since the ratings are typically collected from multiple users, different scoring functions have been employed by existing reputation systems to aggregate various ratings [4, 14, 15, 20, 22, 30]. These systems essentially use a weighting mechanism to incorporate different factors, such as user creditability, personal preference, timeliness of the ratings, to determine the trustworthiness.

A central component of the proposed GCCR framework is a user-centric strategy that exploits the historical user and media provider interactions as evidence to assess the trustworthiness of different media providers. Hence, GCCR essentially follows the feedback mechanism used by existing reputation systems. Since there currently lacks a rating mechanism for users to explicitly provide their feedbacks, GCCR exploits the implicit user feedbacks that are captured by the following relationships in a heterogeneous social network. Intuitively, if a user follows a media provider, it implies that the user still *trusts* the content from that provider. Besides the implicit feedback information, heterogeneous social networks also posses special properties that can be leveraged for trust-aware media recommendation. More specifically, Figure 3 shows the subscription relationships distribution of two datasets crawled from Sina Weibo, the larger one consists of 5,353 user nodes and 581 media nodes whereas the smaller one consists of 500 user nodes and 50 media nodes. From Figure 3a, it can be discovered that only 20 % of user nodes follow more

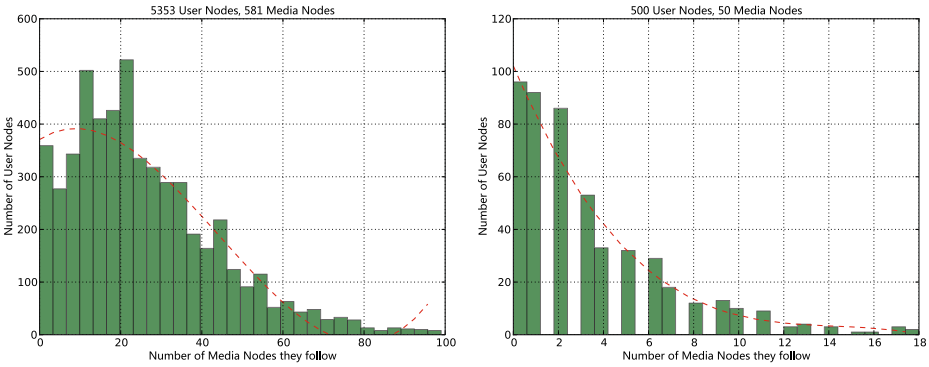


Figure 3 Subscription relationship distribution of two datasets crawled from Sina Weibo

than 10 % of media nodes, and most user nodes only follow less than 5 % of media nodes.

The long tail of the distribution corresponds to the large number of cold-start users, which are either new users or users who are not frequent visitors of the SNS. As the system does not have much knowledge about the cold-start users, their subscriptions with the media nodes are usually unreliable and less predicable. Hence, media nodes that are only followed by cold-start users are typically less trustable than those that are followed by warm-start users, which the system has a relative complete profile constructed through their regular interactions with other media nodes. This premise leads to a two-phase trust-aware process that provides accurate media recommendation for users of heterogeneous social networks. In particular, the first phase extracts a dense subset from a sparse network structure, which corresponds to the warm-start users along with the media providers they follow. These users are then clustered so that users with similar interests (i.e., following similar media providers) are grouped together. Since these subsets of users and media providers are deemed as more trustable and reliable, the user clusters generated in the first phase will be used as seeds to cluster the cold-start users in the second phase. Since the cold-start user nodes do not have many subscription relationships with the media nodes, we propose a content based approach to assign these users into the user clusters formed in phase one. In particular, users' posts in the SNS will be used to construct content vectors from which the similarity between a cold-start user and a user cluster can be computed. The two-level cluster structure allows media recommendation to be performed in a trust-aware fashion. For example, media nodes followed by warm-start users can be recommended ahead of the media nodes that are only followed by cold-start users.

It is worth to note that the structure of a heterogeneous social network allows to recommend both one-way subscription relationship and two-way friendship to a user. For example, it may recommend Jackie to Jim as Jackie is Jim's classmate, and recommend "NBA" to Jim as Jim is interested in basketball. In this paper, we will primarily focus on providing trust-aware recommendations on the subscription relationship for social network users, which has rarely been addressed in existing social network literatures.

The main contribution of this paper can be summarized as follows:

- We propose a trust-aware social media recommendation framework, refereed to as GCCR, to recommend subscription relationship for social network users, which has rarely been addressed in state-of-the-art social network research works.
- A two-phase mechanism is proposed to tackle the cold-start problem and guarantee the recommendation quality, in which graph summarization, content-based approach, clustering, and recommendation algorithms are employed.
- To evaluate the performance of the proposed GCCR framework, we crawl real data from Sina Weibo. Comprehensive experiments demonstrate the effectiveness of GCCR.

The remainder of this paper is organized as follows. We give an overview of existing works that are most relevant to the proposed framework in Section 2. We show the architecture of GCCR and present the key algorithms in Sections 3 and 4, respectively. We assess the effectiveness of the proposed framework in Section 5 and conclude in Section 6.

2 Related work

Recommender Systems are information processing systems that actively gather various kinds of data in order to build their recommendations [23]. Collaborative filtering has emerged as a key technology adopted by many modern recommendation system [1, 2, 9, 11, 12]. It predicts the active user's preference on an unknown item based the feedbacks of other users. Most existing collaborative filtering approaches fall into two categories: neighborhood-based and model-based. The neighborhood-based approaches are further divided into user-based [11] and item-based approaches [19, 25]. The intuitive idea is to identify *similar* users with the active user and compute predictions based on the feedbacks of these similar users. The *similarity* between two users is measured based on the feedbacks on the common items. The neighborhood-based approaches suffer from the feedback scarcity issue that arises in practice because a typical user may only provide feedbacks for a limited number of items. This is even more serious in a heterogeneous social network as a large number of users only follow very limited number of media providers. Users have to follow at least two common media providers in order to be considered as similar. In this regard, only very limited information can be used, which will lead to poor recommendation result. Model-based approaches alleviate the feedback scarcity issue by generating a global model based on the given training data and use the model to predict the active user's preference on the unknown items. Typical models include aspect models [12], latent factor models [2], Bayesian models [33], and decision trees [1]. A major issue with the existing model-based approaches is their high computational overheads which are caused by the tuning of a large number of parameters embedded in the models. This makes it hard to apply these models into large-scale social networks.

With the population of SNSs and the explosive growth of information sources, recommendation systems and related techniques have been increasingly adopted to support decision making by effectively leveraging the social network structure captured by the SNSs. For instance, Spertus et al. present an approach to recommend online communities to users based on their current social network community

membership and evaluate several different similarity measures in a large-scale study using the social networking site Orkut [28]. Geyer et al. leverage social network information for topic recommendation, which outperforms content matching based approaches [8]. Groh et al. propose to improve the classical collaborative filtering methods by deriving user neighborhood information from social network structures [10]. Content-based recommendation technique has also been applied in SNSs. For example, Scott Piao et al. exploit a set of natural language processing tools to develop a real-time system for automatic user interest extraction and make recommendation based on users with similar interests [26]. The proposed framework GCCR goes beyond the existing approaches by addressing the trust issue to provide trust-aware social media recommendation to social network users. Furthermore, the two-level cluster mechanism will more effectively address the cold-start issue, which could not be appropriately addressed by most existing approaches.

Another key difference between the proposed GCCR framework and existing social network recommendation systems lies in the type of recommendations they provide. Existing approaches primarily focus on recommending the two-way friendship relationship based on the social network structure [13, 16, 24]. In contrast, our approach focuses on one-way subscription relationship recommendation, which is to recommend social media to network users.

3 Framework overview

Figure 4 gives an overview of the proposed social media recommendation framework. The framework consists of four key components: data pre-processing,

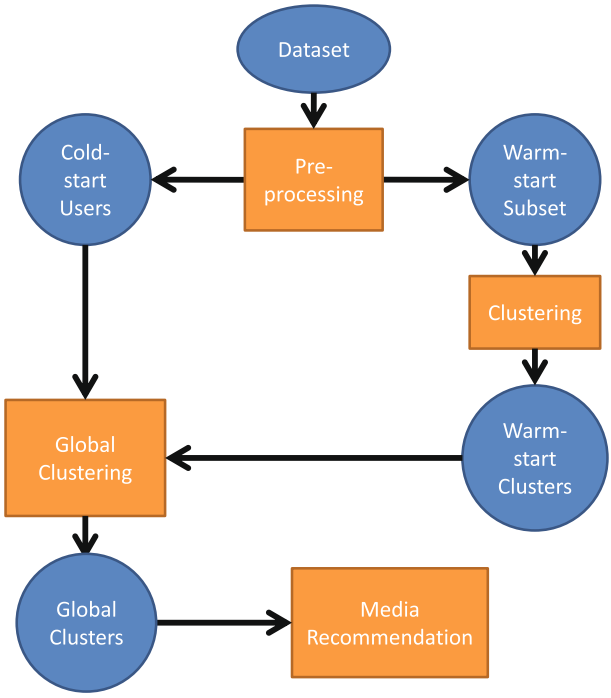


Figure 4 Framework overview

warm-start subset clustering, global clustering, and media recommendation. Details are elaborated as follows:

1. **Data pre-processing** constructs user's interest matrix based on the following relationship between user nodes and media nodes. User nodes will be classified into *warm-start subset* and *cold-start subset* based on the number of media nodes they follow.
2. **Warm-start subset clustering** clusters user nodes of the warm-start subset according to their interests by employing the graph summarization approach (see Section 4.1).
3. **Global clustering** clusters user nodes of the cold-start subset based on the warm-start subset clustering results. More specifically, for each cold-start user node, its content vector is extracted from the tweets the user posted. The content vector of each warm-start cluster is extracted in the same way. A cold-start user node will be assigned to the warm-start cluster with the most similar content vector (see Section 4.2).
4. **Media recommendation** exploits the *Slope One* algorithm [18] to make the final media recommendation based on the global clustering result. (see Section 4.3).

It should be noted that steps 1, 2, and 3 can all be processed off-line, which enables the system to efficiency process new users' recommendation requests. More specifically, we just need to calculate the similarity between the new users and other global clusters, add the user to the most similar cluster, and implement step 4. With the separation of off-line calculation and online recommendation, the efficiency of media recommendation can be guaranteed.

4 GCCR

Given a heterogeneous social network, which consists of N user nodes and M media nodes, it can be represented as user set $U = [u_1, u_2, u_3, ...u_N]$ and a media set $S = [s_1, s_2, ..., s_M]$. For each user node u_i, we model its interest as a vector $v_i = (a_{i1}, a_{i2}, ..., a_{iM})$, where a_{iM} represents u_i's interest to media s_M. In this way, the interest of the whole user set can be formed as a $N * M$ interest matrix m. For user u_i and media s_j, if u_i has followed s_j, the corresponding value in matrix $a_{ij} > 0$, otherwise $a_{ij} = 0$. To discriminate cold-start users from warm-start users, we set a metric interest density, denoted as $des(u_i)$, to evaluate the activity of user u_i in the social network. In particular, the value of $des(u_i)$ is the proportion of media nodes that u_i follows in the whole media nodes. For example, given 100 media nodes, $des(u_i) = 8$ % if u_i only follows 8 media nodes. Given a density threshold λ, the user with interest density higher than or equal to λ is denoted as a *warm-start* user, and as a *cold-start* user if otherwise. Hence, the *warm-start* user subset U' is denoted as $U' = \{u_i | u_i \in U \wedge des(u_i) \geq \lambda\}$.

Based on the above interest matrix m, we generate a directed user interest graph $G(V, E)$, in which V is the collection of user nodes and media nodes, that is, $V = U \cup S$. E is the edge set, which captures the subscription relationship: $E = \{(u_i, s_j) | u_i \in U \wedge s_j \in S \wedge a_{ij} > 0\}$. Similarly, the interest matrix of *warm-start* subset is denoted as m', and the corresponding interest graph is denoted as $G_{m'}$.

4.1 Warm-start subset clustering

Large graph datasets are ubiquitous in many domains, including social networking and biology. Graph summarization techniques have been widely used in such domains as they are effective in discovering useful patterns that are hidden in the underlying data. One important type of graph summarization is to produce small and informative summaries based on user-selected node attributes and relationships. As an example, an interactive graph summarization approach, called *K-SNAP*, is developed that allows users to control the resolutions of summaries and provides the *drill-down* and *roll-up* abilities to navigate through summaries with different resolutions [32]. Inspired by the existing graph summarization techniques, especially the *K-SNAP* algorithm, we propose the *SNAP-Cluster* algorithm to cluster *warm-start* user subset based on the *warm-start* interest graph. It is worth to note that the reason that we don't directly apply *SNAP-Cluster* on the whole user set is that the sparse interest matrix will lead to low clustering accuracy, while the rich information in *warm-start* subset helps guarantee good clustering performance. In what follows, we first introduce two important parameters, *ambiguity* and *diversity*, used by the proposed clustering algorithm in Section 4.1.1. Then we give the details of the *SNAP-Cluster* algorithm in Section 4.1.2.

4.1.1 Key algorithm parameters

Given a *warm-start* subset interest matrix m' and the corresponding interest graph $G_{m'}$, assume that this subset has been clustered into a set of user clusters C_i: $U' = \bigcup C_i$, where $C_i \neq \phi$ and $C_i \cap C_j = \phi \; \forall i \neq j$. Given a media node s_j, the contribution of C_i to s_j is defined as follows.

$$P_{s_j}(C_i) = \{u | u \in C_i \wedge (u, s_j) \in E\} \tag{1}$$

$P_{s_j}(C_i)$ measures the number of users in C_i that follow media node s_j. In particular, the contribution degree of C_i to s_j, denoted as p_{ij}, is defined as follows:

$$p_{ij} = \frac{|P_{s_j}(C_i)|}{|C_i|}, \tag{2}$$

where $|C_i|$ denotes the number of users in cluster C_i, and $|P_{s_j}(C_i)|$ denotes the number of users in C_i that follow s_j. Intuitively, a larger contribution degree implies a higher popularity of s_j in C_i. By setting a contribution degree threshold σ, we claim that *cluster* C_i *strongly follows media* s_j if $p_{ij} \leq \sigma$.

The ambiguity of user cluster C_i to media node s_j is defined as:

$$Amb_{ij} = \begin{cases} |C_i - P_{s_j}(C_i)| & \text{if } p_{ij} \geq \sigma \\ |P_{s_j}(C_i)| & \text{if } p_{ij} < \sigma \end{cases}$$

Furthermore, we define ambiguity of C_i to a collection S of media nodes as: $Amb_i = \sum_{s_j \in S} Amb_{ij}$. Thus the global ambiguity of user collection U' to media collection S is defined as follows:

$$Amb = log\left(\frac{\sum_{C_i \in Clus} Amb_i}{|Clus|}\right), \tag{3}$$

where *Clus* is the set of clusters and the log operator is employed to ensure that global ambiguity decreases linearly with the number of clusters.

As cluster C_i is a set of user nodes, the interest of C_i to media node s_i can be treated as combination of each individual user's interest to s_i. However, if there are only few users in C_i that follow s_j, it implies that C_i shows no strong interest in s_j. Hence, the interest of C_i to s_j, denoted as ca_{ij}, is defined as:

$$ca_{ij} = \begin{cases} \frac{\sum_{u_k \in C_i} a_{kj}}{|C_i|} & \text{if } p_{ij} \geq \sigma \\ 0 & \text{if } p_{ij} < \sigma \end{cases} \tag{4}$$

This allow us to define the interest vector of C_i to the media collection as $cv_i = (ca_{i1}, ca_{i2}, ..., ca_{iM})$, in which each non-zero entry represents a strongly following relationship. Furthermore, we use the cosine distance between interest vectors to measure the difference of clusters' interest, as the following shows:

$$diff(C_i, C_j) = \frac{cv_i \cdot cv_j}{|cv_i| \cdot |cv_j|}, \tag{5}$$

where $diff(C_i, C_j)$ represents the interest difference between C_i and C_j. Thus the diversity of a cluster collection *Clus* derived from user collection U' on media collection S is defined as follows:

$$dvst = \frac{\sum_{C_i \in Clus, C_j \in Clus} diff(C_i, C_j)}{|Clus|} \tag{6}$$

4.1.2 Graph summarization algorithm

Based on the previous definitions, we present the details of the proposed algorithm *SNAP-Cluster* in Algorithm 1. Specifically, the algorithm runs k iterations and each iteration produces a clustering result *Clus*, which is initialized as the warm-start user

Algorithm 1 SNAP-Cluster Algorithm

Input: *warm-start* subset U', media collection S, *warm-start* interest graph $G_{m'}$
Output: *warm-start* user cluster collection *Clus*

1: $Clus = U', maxAmb = 0, srcCi = null, arget = null$
2: **while** $Amb = 0$ or the number of iterations reaches k **do**
3: **for** C_i in *Clus* **do**
4: calculate Amb_{ij} for s_j, find $s_j = argmax(Amb_{ij})$
5: calculate Amb_i for media collection S
6: **if** $Amb_i > maxAmb$ **then**
7: $maxAmb = Amb_i$
8: target=s_j
9: srcCi=C_i
10: **end if**
11: **end for**
12: delete cluster C_i from *Clus*
13: $C_i^{'}, C_i^{''} = split(C_i, target)$ // split cluster
14: add $C_i^{'}$ to *Clus*
15: store current clustering result *Clus* and corresponding Amb, $dvst$ value
16: **end while**
17: **return** optimal *Clus*, which makes the value of $Amb \cdot dvst$ maximum

Algorithm 2 Cluster Split Algorithm

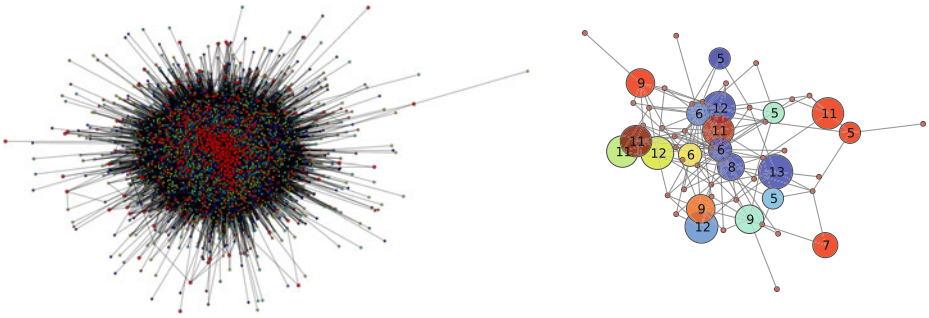
Input: user cluster C, media node s_j, interest graph G
Output: contribute collection $C_i^{'}$ and non-contribute collection $C_i^{''}$ of user cluster *Clus*

1: initialize $C_i^{'}$ and $C_i^{''}$ as null
2: **for** u_i in C **do**
3: **if** $(u_i, s_j) \in E(G)$) **then**
4: add u_i to $C_i^{'}$
5: **else**
6: add u_i to $C_i^{''}$
7: **end if**
8: **end for**
9: **return** $C_i^{'}$

set U'. For each C_i in *Clus*, we calculate the ambiguities for media nodes and pick the media node s_j with the highest ambiguity (lines 3–11 in Algorithm 1). Then we remove cluster C_i and split it according to media node s_j, as shown in Algorithm 2. For each user $u_i \in C_i$, if u_i follows media node s_j, we add user u_i to the contribute collection $C_i^{'}$ (lines 3–4 in Algorithm 2).

After splitting the cluster C_i, we get contribution collection $C_i^{'}$ and non-contribution collection $C_i^{''}$. Then we add the contribution collection $C_i^{'}$ to *Clus* and store the clustering result *Clus* along with the *Amb* and *dvst* values (lines 12–15 in Algorithm 1). After k iterations, we get k different clustering results. Finally, we pick the clustering result that achieves the maximum value on $Amb \cdot dvst$ as the final result (line 17 in Algorithm 1).

It can be observed that the larger the diversity is, the greater the difference between clusters is. Hence, if the cluster interest feature is more obvious, the interest prediction is more precise. In contrast, larger ambiguity implies that users in the cluster have more different interests, which helps improve the recommendation diversification. The proposed *SNAP-Cluster* algorithms aim to achieve a clustering result that maximize the overall effect of diversity and ambiguity. Figure 5a shows



(a) before clustering (b) after clustering

Figure 5 User interest graph

the interest graph generated from about 500 users and 50 medias (average interest density is 7.2 %). Figure 5b is the interest graph after running *SNAP-Cluster* (average interest density is 15 %, small nodes are media nodes, big nodes are user clusters, and the numbers represent the cluster sizes). Thus, employing *SNAP-Custer* helps to improve the recommendation performance.

4.2 Global clustering

In this step, we aim to combine the *cold-start* users and the clustering results of *warm-start* users to complete the task of global clustering. Intuitively, global clustering is achieved by by measuring the interest similarity between individual *cold-start* users and the *warm-start* clusters and assigning a *cold-start* user to its closest cluster. Nonetheless, we can no longer leverage subscription relationship between the *cold-start* user nodes and media nodes to extract their interest vectors due to data sparsity. To tackle this issue, we propose to use a content-based approach to construct *cold-start* users' interest vectors. More specifically, we extract useful keywords from a user's tweets on the SNS and apply TF-IDF to compute the content-based interest vector. Given a *cold-start* user u_i, the interest vector of u_i is denoted as $V_{u_i} = (w_1, w_2, ..., w_K)$. Similarly, the interest vector of a warm-user cluster C_j can be constructed by extracting keywords from the tweets posted by the warm-start users in C_j.

We employ NGD (Normalized Google Distance) [3] to compute the interest similarity between a *cold-start* user and a warm-user cluster. More specifically, the similarity between u_i and C_j, denoted as Sim_{u_i,C_j}, is defined as:

$$Sim_{u_i,C_j} = \frac{\sum_{w_p \in V_{u_i}} \sum_{w_q \in V_{C_j}} sim(w_p, w_q)}{|V_{u_i}| \times |V_{C_j}|}, \tag{7}$$

where $|V_{u_i}|$ means the dimension of interest vector V_{u_i}, and the equation for computing the similarity between two words is defined as:

$$sim(w_p, w_q) = 1 - NGD(w_p, w_q) \tag{8}$$

In (8), we compute the similarity between two words using NGD based on the word co-existence on the Web. More specifically, the NGD between two terms w_p and w_q is calculated as follows:

$$NGD(w_p, w_q) = \frac{max\{\log f(w_p), \log f(w_q)\} - \log f(w_p, w_q)}{\log N - min\{\log f(w_p), \log f(w_q)\}}, \tag{9}$$

where $f(w_p)$ denotes the number of pages containing w_p, $f(w_p, w_q)$ denotes the number of pages containing both w_p and w_q, and N is the total number of web pages searched by Google.

4.3 Media recommendation

After the global cluster collection *GClus* is obtained, we calculate user cluster C_i's interest vector over media collection S, denoted as $cv_i = (ca_{i1}, ca_{i2}, ..., ca_{iM})$, using (4). Then we generate a global interest matrix $\overline{m}$ by combining all user clusters' interest vectors to media collection S. The size of global interest matrix is

$|GClus| * M$. To improve the accuracy of media recommendation, we implement a data-smoothing process on the global interest matrix by replacing the zero value in the matrix with a predicted one. The *Slope One* algorithm [18] is employed to realize the prediction work. First, we define the interest deviation between cluster C_i and C_j as:

$$dev_{C_i,C_j} = \sum_{s_k \in S} \frac{ca_{ik} - ca_{jk}}{|S|} \tag{10}$$

Then, for each zero value in the i^{th} row, e.i., $cv_i = (ca_{i1}, ca_{i2}, ..., ca_{iM})$, we replace it with the predicted value calculated by the following equation.

$$Pre_i = \widehat{cv_i} + \frac{\sum_{j=1}^{M} dev_{C_j,C_i}}{M-1}, \tag{11}$$

where $\widehat{cv_i}$ is the average value of every component of cv_i. In this way, all entities with a zero value in the global interest matrix will be replaced with the predicted values.

As for the recommendation, we first find the cluster that the target user belongs to and then sort the corresponding cluster interest vector in the global interest matrix to generate the top-k recommendations. In particular, the media nodes that the target users has followed should be returned. Furthermore, the media nodes that are followed by warm-start users may also be regarded as more trustworthy than the nodes followed only by cold-start users.

5 Experiments

We implement a set of experiments to evaluate the effectiveness of the proposed media recommendation framework. The experiments are conducted on Windows 7, with an Intel Core 2 Duo CPU at 2.53GHz and 4GB RAM. Experiment code is implement in Python 2.6 and Java with jdk 1.6.

5.1 Experiment setup

Although real-world social network data is used in our experiments, no user interest value is directly available in any real datasets. Hence, we need a method to measure the users' interest to media nodes. Specifically, we use *expected repost ratio* to measure user's interest to media. It can be interpreted as *the probability of a user to repost or comment the tweets of a media*. The expected repost ratio can be calculated using the following equation:

$$a = \frac{P(r|R)}{P(r)} = \frac{P(r)P(R|r)}{P(R)P(r)} = \frac{P(R|r)}{P(R)}, \tag{12}$$

where $P(R)$ is the probability of a user reading the media's tweets, and $P(R|r)$ is the probability of the tweets the user reposted or commented is posted by the media. $P(R)$ and $P(R|r)$ can be calculated from the given dataset.

We crawled the data from the Sina Weibo site [27]. As randomly crawling data will make the dataset too sparse, we simulate the formation of weak-relation based SNS by starting with a small set (i.e. 10) of seed users and expending the dataset by

following their following and follower links. For each user we also download up to 100 recent tweets. The whole process can be described as following:

1. Use 5–10 adjacent users as seed nodes.
2. For each iteration, we use *DFS* to crawl the adjacent user nodes of the current user and use *BFS* to crawl the media nodes the current user has followed.
3. Dynamically adjust the crawling ratio of user nodes and media nodes, according to the current ratio of user nodes and media nodes.
4. Get the reposting and commenting data from the crawled user collection and media collection. Then we calculate the *expected repost ratio*, which represents the user's interest to media. Finally, we can get the user-media interest matrix.

In Sina Weibo, the media nodes include public accounts that are verified by the site and some unverified accounts whose follower/followee ratio is very large. In our experiment, we crawled multiple set of user-media interest matrices and the final experiment result is the averaged result over these multiple datasets. Every dataset includes about 500 user nodes, 50 media nodes, and about 20,000 tweets.

For the sake of performance comparison, we also implemented the following two algorithms on the crawled data: (1) User-based top-k collaborative filtering (CF), and (2) Content-based recommendation. It should be noted that the source codes of these two algorithms are both based on the open source machine learning library Apache Mahout. The user-based algorithm adopts Pearson Correlation Coefficient (PCC) to measure the user similarity. The content-based algorithm is implemented based on the content of tweets that media nodes posted, and we adopt (7–9) to compute the similarity of tweets.

5.2 Evaluation metrics

In our experiment, we select a test user, hide some of her subscriptions, and apply the algorithms to recommend a set of medias to the user. Table 1 shows four possible outcomes based on the recommended and true (i.e. hidden) medias.

We count the number of examples that fall into each classification in Table 1 and compute the following quantities to evaluate the performance of media recommendation:

$$Precision = \frac{\#tp}{\#tp + \#fp}, \quad Recall = \frac{\#tp}{\#tp + \#fn} \tag{13}$$

A metric that combines precision and recall is the harmonic mean of precision and recall, a.k.a., the traditional F-measure or balanced F-score:

$$F = \frac{2 * precision * recall}{(precision + recall)} \tag{14}$$

Table 1 Classification of the possible result of a recommendation of a user to a media

	Recommended	Not recommended
Used	True-positive (tp)	False-negative (fn)
Not used	False-positive (fp)	True-negative (tn)

It should be noted that precision and recall are both single-value metrics based on the whole list of medias returned by the system. For systems that return a ranked sequence of medias, it is desirable to consider the order in which the returned medias are presented. In this paper, we employ a wildly accepted metric, Average precision(AP), which emphasizes ranking relevant medias. It is the average of precisions computed at the point of each relevant media in the ranked sequence:

$$MAP = \frac{\sum_{k=1}^{U} AP(k)}{U} \tag{15}$$

The last metric we use for evaluation is the diversity of media recommendation. Diversity is generally defined as the opposite of similarity. In some cases, suggesting a set of very similar or almost identical items may not be as useful for a user. As an example, given a system that recommends vacation packages. Presenting a list with 5 recommendations, all for the same location, varying only on the choice of hotels or the selection of attractions, may not be as useful as suggesting 5 different locations. Similarly, in media recommendation, if a user likes basketball, recommending other relevant sports may be better than just suggesting *NBA* or *Yao Ming*. Hence, we consider diversity as an important quality factor when evaluating different recommendation algorithms.

5.3 Performance comparison

In this section we compare the performance of GCCR, User-based Collaborative Filtering approach and Content-based approach for media recommendation. In particular, we compare the recommendation results of the above three algorithms based on datasets with different data density.

We first compare overall recommendation accuracy which measures the ratio between the actual recommended media number and the expected recommended media number. As shown in Table 2, GCCR clearly outperforms the other two algorithms in all cases. It can be discovered that when data is extremely sparse, CF and Content-based algorithms cannot generate enough recommendations while GCCR is more robust to data sparsity. That is, GCCR approach is more applicable, as the real scenario of recommendation in SNS is always very spare.

Figure 6 reports the performance comparison of above three media recommendation approaches in terms of *precision*, *recall*, *F-measure*, and *MAP*. From Figure 6a, it can be discovered that the proposed GCCR approach outperforms the other two approaches in terms of *precision* while the density varies. In particular, we can find that when data density is less than 10 %, CF has the lowest precision and the precision of CF clearly increases with the data density. Content-based method is not

Table 2 Performance comparison of recommendation accuracy

Data density	Actual recommended medias/expected recommended medias			
	CF (Top-4) (%)	CF (Top-10) (%)	Content-based (%)	GCCR (%)
5.5	7.1	39.8	16	**99.3**
6.75	8.1	41.9	18.1	**99.7**
8.65	8.4	43.8	20.7	**100**
9.75	8.8	45.6	18.9	**100**

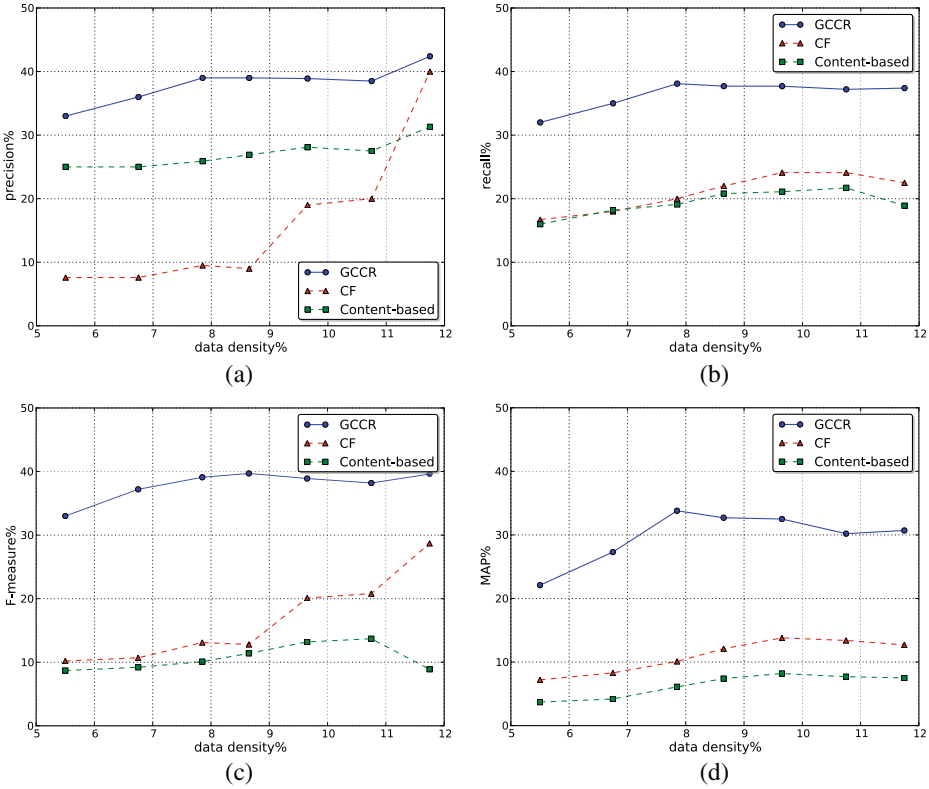


Figure 6 Performance comparison of three approaches

sensitive to data density, while the precision stays low. GCCR achieves relatively high precision even for very sparse data, which is due to the effectiveness of graph summarization. And the precision of GCCR also increases with the increase of data density. The effectiveness in dealing with sparse data makes GCCR especially suitable for making recommendation in heterogeneous Microblogging sites, where data sparsity commonly exists with a density usually less than 10 %. Figure 6b shows the performance comparison of the three media recommendation approaches in terms of recall. It can be discovered that GCCR has the best performance in terms of *recall*, while CF outperforms content-based approach. The performance comparison results in terms of *F-measure* and *MAP* show the similar results as given in Figure 6c and d, respectively.

5.4 Recommendation diversity

To evaluate the performance of recommendation diversity, we first manually classify the media nodes into 15 categories, such as movie, fashion, literature, sports and so on. Cold-start users are new users who have followed only few media nodes. Using similarity based approaches is unlikely to find similar users since the cold-start users only have very limited subscriptions. In this experiment, we delete one or more media

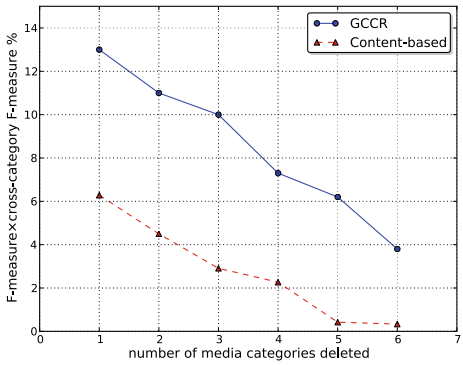


Figure 7 Performance comparison in recommendation diversity

categories that user have followed to simulate the cold start scenario. In particular, these deleted categories are defined as the testing set, while the remaining part is defined as the training set.

Given a user u, if he hasn't follow any media in category C and we recommend one media in category C to him, then we say it's a *cross-category recommendation*. If the cross-category media is in the testing set, we say it's a *cross-category hit*. In this paper, we use *cross-category recommendation* F-measure (similar to (14)) to evaluate the recommendation diversity, as Figure 7 shows. It can be discovered that GCCR largely outperforms the content-based approach with the vary of testing set volume in terms of diversity, which is largely caused by the clustering ambiguity.

5.5 Effect of parameters

Ambiguity Ambiguity is the measurement of the internal subscription difference between users in the target cluster. It is an important parameter employed in the process of graph summarization. From Figure 8a, we can find that the overall ambiguity of the current clustering result decreases with the increase of cluster numbers. This is because when cluster number is small, it's easier to form strongly following relationships. The recommendation result gets better with the decrease of ambiguity, which is more obvious when data density is large. When the cluster number is too large, the recommendation accuracy gets lower. As diversity will increase at the expense of other properties, such as accuracy [31], we compute curves to evaluate the decrease in accuracy vs. the increase in diversity.

Diversity Diversity reflects the external interest difference between clusters, which increases with the number of clusters. In Figure 8b, it can be discovered that global recommendation diversity decreases with the increase of *dvst*. When the number of cluster is 10, *dvst* reaches the lowest value (e.g., 0.524), and the recommendation diversity performance of GCCR reaches the highest value. When the number of clusters is 80, the value of *dvst* reaches the highest point, while the diversity performance is the worst. This could be explained that, when the difference between external interests increases, the difference between internal subscription decreases, which finally leads to the decrease of recommendation diversity.

From the above experiments, we can find that the number of clusters clearly affects the performance of recommendation. More clusters lead to the decrease

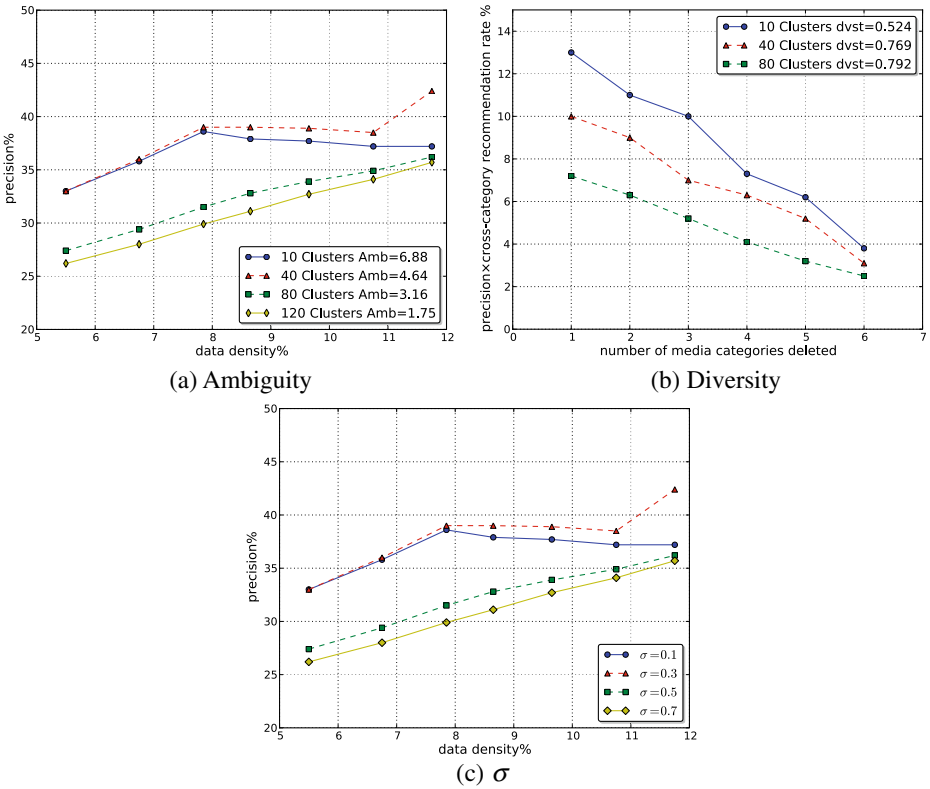


Figure 8 Evaluation of parameter effects

of ambiguity of individual clusters, which improves the recommendation precision but decreases the recommendation diversity. In contrast, less clusters will generate more diverse recommendation result, and improve the recommendation precision in cold-start scenario. It should be noted that the choice of clustering strategy depends on specific recommendation requirements. In practice, with no specific recommendation requirements, we choose the clustering strategy which makes the greatest product of ambiguity and diversity.

Contribution degree threshold σ σ defines the minimum coverage of strongly following relationships. When we need to judge whether a cluster C has interest in a media m, we use σ as the threshold. If σ is large, the cluster should have many users who have subscription relationships with media m. In contrast, when σ is small, it's easier to form strongly following relationships. Figure 8c shows the effect of σ on recommendation precision.

As for the choice of optimal σ, σ is set as 0.5 in Tian's paper [32]. However, for extreme sparse datasets, a relatively loose threshold for strongly following relationships is better (e.g., $\sigma = 0.3$ is optimal according to our experiment). Because a low threshold will generate more non-zero parts after graph summarization, the optimal σ also depends on the actual dataset.

6 Conclusion

We present a novel framework for media recommendation in heterogeneous social networks. The framework exploits a two-phase process to provide accurate and trust-aware recommendations for social network users. The two-phase process integrates graph summarization and content-based clustering to effectively address the data sparsity issue, which makes it especially suitable for making social network recommendations where data sparsity commonly exists. Experimental results on real-world social network data demonstrate that the proposed framework clearly outperforms other recommendation algorithms in terms of *precision*, *recall*, *F-measure*, *MAP*, and *diversity*. In addition, the performance gap between GCCR and other approaches is more obvious in cold-start scenarios. In our future work, we plan to deploy our media recommendation system onto the Sina Weibo site. This will allow us to collect users' feedbacks on our recommendation results, which will be helpful for us to adjust the recommendation strategies.

Acknowledgements This research was partially supported by the National Technology Support Program under grant of 2011BAH16B04, the National Natural Science Foundation of China under grant of No. 61173176, Science and Technology Program of Zhejiang Province under grant of 2008C03007, Zhejiang Provincial Natural Science Foundation of China under grant number Y1110591, National High-Tech Research and Development Plan of China under Grant No. 2011AA010501.

References

1. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: UAI '98: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43–52. Morgan Kaufmann (1998)
2. Canny, J.: Collaborative filtering with privacy via factor analysis. In: SIGIR '02: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 238–245. ACM, New York (2002)
3. Cilibrasi, R.L., Vitnyi, P.M.B.:The google similarity distance. IEEE Trans. Knowl. Data Eng. **19**(3), 370–383 (2007)
4. Conner, W., Iyengar, A., Mikalsen, T., Rouvellou, I., Nahrstedt, K.: A trust management framework for service-oriented environments. In: WWW '09: Proceedings of the 18th International Conference on World Wide Web, pp. 891–900. ACM, New York (2009)
5. Dellarocas, C.: The digitization of word of mouth: Promise and challenges of online feedback mechanisms. Manag. Sci. **49**(10), 1407–1424 (2003)
6. Douban: http://www.douban.com/. Accessed 14 Jul 2013
7. Facebook: http://www.facebook.com/. Accessed 14 Jul 2013
8. Geyer, W., Dugan, C., Millen, D.R., Muller, M., Freyne, J.: Recommending topics for self-descriptions in online user profiles. In: RecSys'08: Proceedings of the 2008 ACM Conference on Recommender Systems, pp. 59–66 (2008)
9. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–70 (1992)
10. Groh, G., Ehmig, C.: Recommendations in taste related domains: collaborative filtering vs. social filtering. In: GROUP'07: Proceedings of the 2007 International ACM Conference on Supporting Group Work, pp. 127–136 (2007)
11. Herlocker, J.L., Konstan, J. A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 230–237. ACM, New York (1999)
12. Hofmann, T.: Latent semantic models for collaborative filtering. ACM Trans. Inf. Syst. **22**(1), 89–115 (2004)

13. John, H., Mike, B., Barry, S.: Recommending twitter users to follow using content and collaborative filtering approaches. In: RecSys'10: Proceedings of the 4th ACM Conference on Recommender Systems, pp. 199–206 (2010)
14. Kalepu, S., Krishnaswamy, S., Loke, S.W.: Reputation = f(user ranking, compliance, verity). In: ICWS '04: Proceedings of the IEEE International Conference on Web Services, p. 200. IEEE Computer Society, Washington(2004)
15. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The eigentrust algorithm for reputation management in p2p networks. In: WWW '03: Proceedings of the 12th International Conference on World Wide Web, pp. 640–651. ACM, New York (2003)
16. Kim, Y., Shim, K.: Twitobi: A recommendation system for twitter using probabilistic modeling. In: ICDM'11: Proceedings of the 11th International Conference on Data Mining, pp. 340–349 (2011)
17. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: WWW '10: Proceedings of the 19th International Conference on World Wide Web, pp. 591–600 (2010)
18. Lemire, D., Maclachlan, A.: Slope one predictors for online rating-based colaborative filtering. In: SDM'05: Proceedings of the SIAM Data Mining, pp. 1–5 (2005)
19. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Comput. **7**(1), 76–80 (2003)
20. Malik, Z., Bouguettaya, A.: Rateweb: Reputation assessment for trust establishment among web services. VLDB J. **18**(4), 885–911 (2009)
21. Myspace: https://www.myspace.com/. Accessed 14 Jul 2013
22. Park, S., Liu, L., Pu, C., Srivatsa, M., Zhang, J.: Resilient trust management for web service integration. In: ICWS '05: Proceedings of the IEEE International Conference on Web Services, pp. 499–506. IEEE Computer Society, Washington (2005)
23. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.): Recommender systems handbook. Springer (2011)
24. Sakaguchi, T., Akaho Y., Takagi, T., Shintani, T.: Recommendations in twitter using conceptual fuzzy sets. In: NAFIPS'10: Proceedings of the 2010 Annual Meeting of the North American Fuzzy Information Processing Society, pp. 1–6 (2010)
25. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW '01: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM, New York (2001)
26. Scott, P., Jon, W.: A feasibility study on extracting twitter users' interests using nlp tools for serendipitous connections. In: SocialCom'11: Proceedings of The Third IEEE International Conference on Social Computing, pp. 910–915 (2011)
27. Sina weibo: http://www.weibo.com/. Accessed 14 Jul 2013
28. Spertus, E., Sahami, M., Buyukkokten, O: Evaluating similarity measures: a large-scale study in the orkut social network. In: KDD'05: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 678–684 (2005)
29. Twitter: https://twitter.com/. Accessed 14 Jul 2013
30. Xiong, L., Liu, L.: Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. IEEE Trans. Knowl. Data Eng. **16**, 843–857 (2004)
31. Zhang, M., Hurley, N., Monotony, A.: Improving the diversity of recommendation lists. In: RecSys'08: Proceedings of ACM International Conference on Recommender Systems, pp. 123–130 (2008)
32. Zhang, N., Tian, Y., Patel, J.M.: Discovery-driven graph summarization. In: ICDE'10: Proceedings of the 2010 International Conference on Data Engineering, pp. 880–891 (2010)
33. Zhang, Y., Koren, J.: Efficient bayesian hierarchical user modeling for recommendation system. In: SIGIR '07: Proceedings of the 30th Annual International ACM SIGIR Conference onJb Research and Development in Information Retrieval, pp. 47–54. ACM, New York (2007)