SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2017, Vol. 60 012102:1–012102:14
doi: 10.1007/s11432-016-5551-7

# Hybrid followee recommendation in microblogging systems

Hanhua CHEN*, Hai JIN & Xiaolong CUI

*Services Computing Technology and System Laboratory, Big Data Technology and System Laboratory, Cluster and Grid Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan* 430074*, China*

Received October 16, 2015; accepted December 7, 2015; published online November 23, 2016

**Abstract** Followee recommendation plays an important role in information sharing over microblogging platforms. Existing followee recommendation schemes adopt either content relevance or social information for followee ranking, suffering poor performance. Based on the observation that microblogging systems have dual roles of social network and news media platform, we propose a novel followee recommendation scheme that takes into account the information sources of both tweet contents and the social structures. We set up a linear weighted model to combine the two factors and further design a simulated annealing algorithm to automatically assign the weights of both factors in order to achieve an optimized combination of them. We conduct comprehensive experiments on real-world datasets collected from Sina Weibo, the largest microblogging system in China. The results demonstrate that our scheme provides a much more accurate followee recommendation for a user compared to existing schemes.

**Keywords** online social networks, microblogging, followee recommendation, simulated annealing, ranking

**Citation** Chen H H, Jin H, Cui X L. Hybrid followee recommendation in microblogging systems. Sci China Inf Sci, 2017, 60(1): 012102, doi: 10.1007/s11432-016-5551-7

## 1 Introduction

Since the emergence of microblogging systems, such as Twitter and Sina Weibo, hundreds of millions of users have become to use the microblogging service as a tool for information sharing on the Internet. For example, as the most popular microblogging system in China, Sina Weibo has attracted more than three hundred million active users[1)]. The community produces more than one hundred million pieces of news (called weibos in Sina Weibo in correspondence to tweets in Twitter) each day. The increase of the population in the Sina microblogging community has been surging sharply by more than 16 million per month. Due to the large population, finding relevant and reliable information for a user in the community becomes a challenge.

To cope with such large-scale information, traditional communities commonly design deliberate recommendation schemes to help users to select information of potential interest [1–4]. For example, an Internet

* Corresponding author (email: chen@hust.edu.cn)

1) http://news.xinhuanet.com/tech/2012-02/29/c_122769084.htm.

 


**Figure 1** "People you may be interested in" in Sina Weibo.

video-on-demand system may use the collaborative filtering (CF) scheme [5] to recommend items using similarities of preference of different users. Such a scheme needs the users' rating information about items. Unlike such systems, a microblogging system serves users in a quite different way. For example, Sina Weibo allows users to post short news. Users follow others or are followed by others. A user gets the updates of all the news posted by the users he/she follows. Thus a user must carefully select other users to follow, so that he/she can benefit most from their weibos. Through the network formed by the followers and followees, Sina Weibo microblogging system provides the users a new platform for information sharing. In such a paradigm, the key problem of information recommendation is how to proactively recommend most relevant followees to a user [6].

Very limited work has been done on followee recommendation in microblogging systems. Traditional widely used recommendation schemes are not applicable to microblogging systems. For example, the CF scheme needs the information such as user's rates, which are difficult to obtain in microblogging followees recommendation.

Most existing microblogging systems assume that a user tends to follow the people with whom he/she has close social relations. The schemes mainly follow the philosophy of online social networks [7–9], such as Facebook, to exploit the social structure information in the system. Figure 1 illustrates the current followee recommendation function in Sina Weibo microblogging platform, named as "people you may be interested in". In their design, candidates followed by more followees of a user, are more likely to be recommended to him/her. Another kind of followee recommendation scheme leverages the interest of users [10]. Hannon et al. [6] make followee recommendation according to the similarity of user profiles that reflect the likely interests of users. The proposed scheme generates the profile for a user using their microblogging histories.

However, a recent research by Kwak et al. [11] based on the trace of the entire Twittersphere, shows that microblogging systems deviate significantly from known characteristics of traditional social networks. Their study indicates that a microblogging system is a platform of both social network and news media. Existing followee recommendation schemes ignore the coexistence of these two features in microblogging systems and may result in poor performance. Based on the unique feature of microblogging systems, we propose a novel followee recommendation scheme in this work. We look at different sources of information available. When recommending candidate followees to a user, our scheme considers both factors of his/her social relations and the content relevance. We set up a linear weighted model to hybridize the two factors. To solve the difficulty in weights optimization in a continuous space of real numbers, we design a simulated annealing (SA) algorithm. The algorithm achieves fast and satisfactory parameter optimization in our model.

We conduct comprehensive experiments using real-world traces collected from the Sina Weibo microblogging system. The results show that our hybrid recommendation scheme greatly outperforms

existing schemes in terms of accuracy.

The main contributions of our scheme are threefold:

• We propose a novel hybrid followee recommendation model based on the unique feature of microblogging systems, which considers both the factors of social structure and content relevance.

• We design an efficient and effective simulated annealing algorithm to optimize the parameter settings in the model.

• We evaluate the performance of this design using real-world traces and demonstrate the great performance improvement by our scheme.

The rest of the paper is structured as follows. In Section 2, we discuss related work. Section 3 presents the hybrid followee recommendation model we proposed. In Section 4, we detail the parameter assignment of content and social information. In Section 5, we evaluate the performance of our design and present the results compared to existing schemes. Section 6 concludes this work with possible future work.

## 2 Related work

As a solution to attention scarcity [12] caused by huge amount of information, recommenders have been studied for years. In diverse applications, different sources of information, such as the preference overlap of users, content relevance and social structure are used to design recommendation algorithms.

One of the most popular approaches is collaborative filtering [13–15]. The scheme makes automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many other users (collaborating). The underlying assumption is that if a user $A$ has the same opinion as a user $B$ on an issue, $A$ is also likely to have $B$'s opinion on a different issue $x$. The users' rates information needed by the CF scheme, however, is difficult to obtain in microblogging followee recommendation.

A second kind of scheme utilizes the content relevance of items [10, 16]. Such recommenders are often applied in domains where extensive textual content of items is available, such as the recommendation of websites [17] and books [18]. For example, to recommend websites, Pazzani et al. [17] create bag-of-word profiles for users from their activities, and then choose websites most relevant to the profiles of the individuals. Hannon et al. [6] exploit the content created by a user and recommend followees to him/her based on the content similarity between the candidate followees and him/her. The implicit assumption of such a scheme is that a target user is likely to follow those who are similar [19], which is consistent with the homophily effect - the principle that we tend to be similar to our friends [20]. Armentanoet al. [10] analyze different content-based profiles of twitter users for followee recommendation. However, making followee recommendation solely considering content relevance will suffer a poor precision of recommendation results [21]. To improve the performance of content-based recommenders, Chechev and Georgiev [22] evaluate several content-based strategies for modeling a Twitter user's profile. They achieve better performance by taking advantage of the availability of more user-generated content including both the free text and hashtags [23] published in users' tweets.

Another kind of scheme leverages the information of social structures [24]. For example, Hill et al. [25] describe a social filtering recommender on Usenet newsgroup. For each newsgroup, they recommend the most frequently mentioned URLs. Andersen et al. [26] propose trust-based recommendation, where they discuss ways to employ users' opinions toward other users to compute recommendations. Shardanand et al. [27] multiply four different algorithms for music recommendations by using social information filtering. Chen et al. [28] propose an approach to recommend interesting URLs coming from information streams using social voting mechanisms. In order to exploit the information source of the social relation for followee recommendation, Armentano et al. [29] consider three factors including the relation between the number of followers and the number of followees, the number of occurrences of each candidate in the final list and the number of friends in common. Golder et al. [30] assume that a user will follow back his or her followers to return the attention and thus introduces the structural approach which considers reciprocity, shared interests, shared audience and filtered people for recommending followees.

Kwak et al. [11] recently identified the dual roles of social network and news media in microblogging

systems based on the empirical study on the trace of the entire Twittersphere. Such a finding casts doubts on the performance of previous followee recommendation schemes which ignore the coexistence of the sources of information of social structure and content relevance. Based on the observation by Kwak et al. [11], in this design, we consider both these two sources of information, and propose a linear weighted model to hybridize the two factors. We tackle the non-trivial problem of parameter optimization in the model by designing a simulated annealing algorithm.

## 3 Followee ranking model

### 3.1 Overview

In this section, we give a mathematical description of our user recommendation model. Borrowing ideas from social networking and SMS messaging, a microblogging system leverages the social network for information sharing. Users follow or are followed by each other. Formally, if user $a$ follows user $b$, we refer to $a$ as $b$'s follower, and $b$ as $a$'s followee. In the follower-followee network, the social structure information is an important resource that can be used to calculate the rank of a user for recommendation [31].

Unlike many other online social networks, the relationships in microblogging systems can be social or informational, or both, because users not only follow others for maintaining social links, but also for gaining access to interesting information generated by others [19]. The most emerging feature of the microblogging system is its content sharing paradigm as a news media [11]. Users of existing microblogging systems, such as Twitter and Sina Weibo can post short messages (weibos/tweets). A microblogging system serves a consumer mainly by polling all his/her followees for gathering all the updates of the messages [32]. Thus it is important for a user to seek and select followees with potential content of interest, so that he/she can benefit most from their tweets.

The major process of our followee recommendation design is as following. The system summarizes the collections of the weibos/tweets recently posted by a user and computes the statistics from the corpus. The social relationship information is also analyzed for a user. When a user logins into the Sina Weibo system, the system recommends the followees to the user based on the text statistics and social structure analysis. Finally, the users which are most likely to be followed are recommended to the user.

To compute the ranking score of followees, our recommendation model takes into account both the content relevance and the social structure information,

$$W = \lambda W_c + (1 - \lambda) W_s, \tag{1}$$

where $W_c$ denotes the normalized ranking score based on content relevance; $W_s$ denotes the normalized score based on social structure information; and $\lambda$ $(0 < \lambda < 1)$ is the parameter scaling the contribution of the factor of content relevance.

### 3.2 Content relevance

We use the *vector space model* (VSM) to rank the content relevance between a user and a followee candidate. A user $u$ in the system is represented using a vector,

$$V_u = (v_u(w_1), v_u(w_2), \ldots, v_u(w_m)), \tag{2}$$

where $(w_1, w_2, \ldots, w_m)$ is a bag-of-word profile for a user, which is created to represent a user's content based on their own tweets. Due to the immanent features of sparsity of information in a single short-text tweet [33], we use a sufficiently large set of recent weibos/tweets posted by a user to create the profile vector for him/her. Other information such as hashtags can also be used in the content-based model [22]. For simplicity, here we only consider the pure text in users' tweets which is the most common case. Specifically, we extract words from $k$ latest tweets posted by a user. In (2), the notation $v_u(w_i)$ quantifies

the strength of interest of the user $u$ in the word $w_i$. To calculate the value of $v_u(w_i)$, we use the most popular TF-IDF term weighting scheme [34],

$$v_u(w_i) = \mathrm{TF}_u(w_i) \times \mathrm{IDF}_u(w_i). \tag{3}$$

The weighting model described in (3) includes two factors, the term frequency (TF) and inverse document (user) frequency (IDF). In a given user's profile, a word's importance increases proportionally to the number of times it appears in the profile and inversely proportionally to its frequency in all users' profiles. A word occurring frequently in a particular user's weibos/tweets but rarely elsewhere in the tweets of other users is thought to be important for that user only. Specifically, TF denotes the term frequency property that is local and content-oriented to a user profile,

$$\mathrm{TF}_u(w_i) = \frac{f_{u,w_i}}{\sum_k f_{u,w_k}}, \tag{4}$$

where $f_{u,w_i}$ quantifies the frequency of appearance of term $w_i$ in the latest tweet list of user $u$. Intuitively, a higher TF of a word means the user mentions the word more frequently, indicating higher interest. The IDF quantifies the fact that terms appearing in more users' tweet profiles are less important,

$$\mathrm{IDF}_u(w_i) = \log\left(\frac{N}{f_{w_i}}\right), \tag{5}$$

where $N$ is the size of the user set, while $f_{w_i}$ is the number of users with the profiles containing $w_i$. A higher score of IDF for a certain word means that the given word can better distinguish one user from others.

Assuming $V_t$ denotes the vector of the target user $u_t$, we compute the content relevance between $u_t$ and the candidate user $u$ using the cosine similarity of the two vectors of them,

$$W_c = \frac{\sum_{w \in V_t} (\frac{f_{u,w_i}}{\sum_k f_{u,w_k}} \times \log \frac{N}{f_{w_i}})}{|V_t| \times |V|}, \tag{6}$$

where $|V_t|$ and $|V|$ are the sizes of the two vectors, respectively.

### 3.3 Social structure

In the following, we calculate the ranking according to the social structure. Assuming we have a list $R$ of candidate users for recommending as followees to a target user $u_t$, we explore several features to give a score to a user $u$ in $R$ to rank them according to social structure information. To consider the information source of social structure, we use some well evaluated features proposed by previous work [29, 35].

The first feature exploited is the ratio of the numbers of followers of user $u$ to the number of users the given user follows [29],

$$w_f(u) = \frac{|\mathrm{followers}(u)|}{|\mathrm{followees}(u)|}, \tag{7}$$

where followers($u$) represents the set of followers of user $u$; followees($u$) denotes the set of followees of $u$.

As shown in (7), this feature inclines to recommend famous or popular users for a target user because a high ratio of the number of their followers to the number of their followees leads to a high score of $w_f(u)$. We take into account this feature based on the recent research results by Garcia et al. [35], which show the promise of recommending users according to the social popularity in a microblogging system.

The second feature corresponds to the number of followers of a candidate user $u$ while the followers who come from target user's one-hop followee set will be taken into consideration [29],

$$w_o(u) = |\mathrm{followers}(u) \cap \mathrm{followee}(u_t)|, \tag{8}$$

where followee($u_t$) represents all the users that the target user directly follows. The fundamental idea behind the equation is the trust propagation principle proposed by Andersen et al. [26], which reveals

the fact that $u_t$'s trust in user $u$ reinforces if the people whom $u_t$ trusts also show their trust in $u$. The larger the value of $w_o(u)$ is, with the higher possibility $u_t$ will be interested in $u$.

The third feature we consider in this model is the number of friends shared between the target user $u_t$ and a certain candidate user $u$ [29],

$$w_t(u) = |\text{followees}(u_t) \cap \text{followees}(u)|. \tag{9}$$

The feature indicates that the more common friends a candidate shares with the target user, the more likely he/she has similar tastes with the target user.

Finally we combine the above three features of social structure information using the following formula,

$$W_s(u) = \alpha w_f(u) + \beta w_o(u) + (1 - \alpha - \beta) w_t(u), \tag{10}$$

where $0 < \alpha < 1$ and $0 < \beta < 1$ $(0 < \alpha + \beta < 1)$ are parameters scaling the contribution of the factors $w_f(u)$ and $w_o(u)$, respectively.

## 4 Weight parameter assignment

As aforementioned, the followee ranking model considers combining different important factors using a linear weighted model. In our hybrid following recommendation model, it is non-trivial to select the configurations of the weight parameters for different factors, as different parameter settings may achieve different performance for recommendation. In practice, it is difficult for a system administrator to manually assign the weights for different factors, as it is hard to know the different importance of diverse factors. This becomes a combinatorial optimization problem for parameter estimation. Formally, combinatorial optimization technique seek to find some configuration of a set of parameters $w = (w_1, w_2, w_3, \ldots, w_n)$ that optimizes some form of objective function, which measures the goodness of a particular configuration of parameters.

Traditional combination search problems could usually be well solved by a number of heuristic algorithms on a finite set. However, existing schemes are not directly applicable to our problem. In our model, the automatic parameter assignment is extremely difficult, since the parameters $\lambda$, $\alpha$, and $\beta$ in (1) and (10) are all in the real number space within the normalized range. The infinite number of combinations make it impossible to perform simple heuristic or exhaustive enumeration to find the global near optimal settings. In order to make the optimization cost-efficient, we need a good search strategy to find a near optimal assignment of the parameters by exploring only a small fraction of the search space compared to the entire search space.

To solve the above problem, we design the SAWA (Simulated Annealing for Weights Assignment) algorithm, which adapts the simulated annealing algorithm [36] to automatically assign the weights of factors in our model. Basically, the SAWA algorithm aims to search for a set of weight parameters which could be used in our recommendation model to produce candidate followees list similar with the list of followees given by a user in a real system. The basic idea of the algorithms is that we keep examining the neighbors of the current best combination of the weight parameters. If a neighbor combination is better, then it will be chosen as the best combination. Different from the heuristics such as the greedy algorithm, the SAWA algorithm deliberately chooses a worse combination occasionally to successfully avoid being trapped in a local optimal area.

Formally, we define the *neighborhood structure* $N(w)$ of a combination solution $w$, which is a subset of the universal set of all the combination solutions. The members of $N(w)$ are close to $w$ in some way. Specifically, the algorithm chooses an initial random assignment as the current assignment $w$. Then the algorithm repeats the following steps until timeout. Each step, the algorithm examines the members in $N(w)$ of the current assignment $w$. If the best neighbor $w^*$ among $N(w)$ achieves better performance than the current assignment, the algorithm chooses $w^*$ to replace the current assignment. If $w^*$ is worse than the current combination, the algorithm still accepts $w^*$ with a low probability.

We find the neighbors of the current combination $w$ by changing the values of the factors. The neighbors of a single factor $w_i$ is experimentally set to $\{w_i - \delta, w_i + \delta\}$. The cost function $f(w)$ is used to measure the quality a combination. It is computed by replacing $w_i$ in the current combination with the value of $w_i - \delta$ or $w_i + \delta$ whichever contributes to a better solution, and applying the new combination to compute the value of the cost function. Here, we define the cost function in (11), which measures the distance between the ranking results and the results provided by the users in the real world system [37],

$$D(A,B) = \frac{\sum_{i=1}^{n}[(n-i) \times \sum_{j=1 \Lambda B_j \notin \{B_1,\ldots,B_i\}}^{i} 1]}{\sum_{i=1}^{[\frac{n}{2}]}[(n-i) \times i] + \sum_{i=[\frac{n}{2}]+1}^{n}[(n-i) \times (n-i)]}, \tag{11}$$

where $A$ and $B$ are test list and recommendation list, respectively; $n$ is the total number of objects in the ranking lists; and $B_i$ denotes the $i$th object in ranking list $B$. In (11) the numerator of the formula is used to quantify the real distance of these two ranking, while the denominator of the formula is used to normalize the real distance to a number between 0 and 1. As we can see, Eq. (11) gives greater penalty to the mismatches for top objects. For example, if the best candidate is ranked wrongly, the weight for the error should be $n-1$, while if only the second best object is ranked wrongly, the weight for the error is $n-2$. It is clear that a smaller list distance represents a better performance. Algorithm 1 describes the SAWA algorithm in detail.

**Algorithm 1** Finding best weights assignment

**Input:** An initial random combination $w_0$;

**Output:** An approximated optimal combination $w$;

**initial** $w = w_0$;
$t = T_0$;
**repeat**
  compute the best neighbor $w^*$ in $N(w)$;
  diff $\leftarrow f(w^*) - f(w)$;
  **if** diff $> 0$ **then** $w \leftarrow w^*$;
  **else**
    generate a random number $x$ in (0,1);
    **if** $x < \exp(-\text{diff}/t)$ **then** $w \leftarrow w^*$;
  $t \leftarrow t - \Delta t$;
**until** $t = 0$;
**return** $w$.

In the algorithm, it is nontrivial to choose the parameter of neighborhood width, i.e., $\delta$, to achieve an advisable tradeoff between the algorithm speed and quality. In practice a smaller value of $\delta$ leads to a more precise approximation to the global optimal solution with more computing time while a large number of $\delta$ cannot precisely discover the global optimal solution. In Section 5, we will experimentally find the setting of $\delta$ which achieves a good tradeoff.

## 5 Experiments

### 5.1 Data sets

We conduct our experiment using the WISE challenge (T2) data set[2)] crawled from Sina Weibo, the most popular microblogging system in China. The data set contains 51.4 million users' social link information and 465 million tweets. Specifically, the dataset includes two data collections:

(1) Tweets collection: basic information about tweets (full-text content, timestamp, user ID, message ID etc.), mentions (user IDs appearing in tweets), re-tweet paths, and whether containing links.

(2) Followship networks: the following network of users based on user IDs.

2) http://www.wise2012.cs.ucy.ac.cy/challenge.html.

### 5.2 Evaluation methodology

In the experiments, we choose the users who at least have 100 followers and 80 followees as our target user set. Further, the set of followees of each target user was partitioned into two parts. One is the fraction of 70% used as training set $S$, starting from which candidates are located and evaluated and finally recommended; while the other is the set $A$ with the remaining fraction of 30% used as test list. For the users in $S$, we use the recommendation algorithm to obtain a recommendation list $B$. If followees in the recommendation list also appear in the test set $A$, it means that the algorithm is able to locate relevant followees.

We implement the SAWA algorithm described in Section 4 to evaluate the performance of our design. In the experiment, we first need to choose the parameter neighborhood width, i.e., $\delta$, to achieve an advisable tradeoff between the algorithm performance and efficiency. It is clear that a large value of $\delta$ cannot precisely discover the global optimal solution, while a too little one will cause the process very time-consuming. Accordingly we adjust the value of $\delta$ from 0.2 to 0.001 and examine both the performance and the efficiency of the algorithm. The results in Figure 2 show that when the value of $\delta$ decreases, the time consumption increases sharply, while Figure 3 shows that after it decreases to the point of 0.05, the quality of the results stops to improve. Thus, we can clearly see that 0.05 is a good granularity for the neighborhood width with a satisfactory performance. Using this setting, we perform the SAWA algorithm and achieve the best settings of the parameters in the algorithm. The results reveal that in our datasets the best parameter settings arrive at $\alpha = 0.039$, $\beta = 0.865$, and $\lambda = 0.142$, which scale the contributions of the factor of social structure $w_f(u)$, $w_o(u)$ and the content relevance $W_c$, respectively. In practice, real world systems can follow the above scheme to achieve the desired best configurations.

In the evaluation, we first use the algorithm solely considering the content-based information or social structure information as the baseline schemes. We further compare the performance of our scheme with the machine learning schemes [38–41] including two existing popular schemes widely used in the IR field, including the PageRank and LDA scheme [42]. We also compare our scheme with Twittomender [6]. In the experiments, we follow the parameter settings for previous designs.

To evaluate the proposed algorithm, we firstly examine our design by two widely used criterions in the field of information retrieval: recall and precision [43, 44]. Recall for a target user $u_t$ is the fraction of relevant followee of $u_t$ that are retrieved.

$$\text{Recall}(u_t) = \frac{\#\text{ of returned relevant followee of } u_t}{\text{total }\#\text{ of relevant followee of } u_t}, \tag{12}$$

where the denominator is the total number of relevant followee of a target user $u_t$, and the numerator is the number of relevant followee of $u_t$ returned by our algorithm. In the experiments, the recall can be computed by $\frac{|B|\cap|A|}{|A|}$.

Precision captures the fraction of relevant followees in the returned results.

$$\text{Precision}(u_t) = \frac{\#\text{ of returned relevant followee of } u_t}{\text{total }\#\text{ of results returned for } u_t}, \tag{13}$$

where the denominator is the total number of followee the algorithm returned for the target user $u_t$.

The ranking accuracy is evaluated using the metric $P@k$ [45], proportion of the relevant results in top-k-ranked results [46].

Although precision and recall measure the overall performance of the presented scheme, one limitation of this measure is that it regards all relevant results equally regardless of where they appear in the list of the top-k recommendation list. For the two cases that the relevant recommendations appear at the top of the ranking using one algorithm and at the bottom of the ranking using the other, the first algorithm performs better even the overall precisions of the two algorithms are similar. In order to better examine the quality of the ranking results, we compute the distance between two ranking lists of the same set of objects. We use the distance defined in (11) [37] to further evaluate the ranking quality.

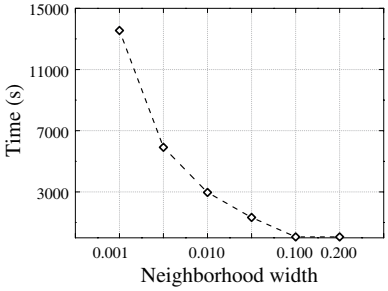


**Figure 2** Factors assignment time for different $\delta$.

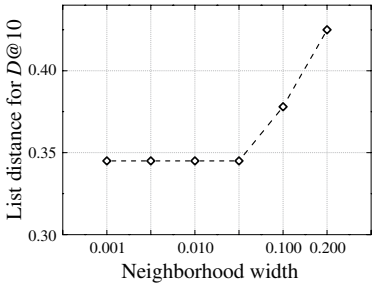


**Figure 3** List distance for different $\delta$.

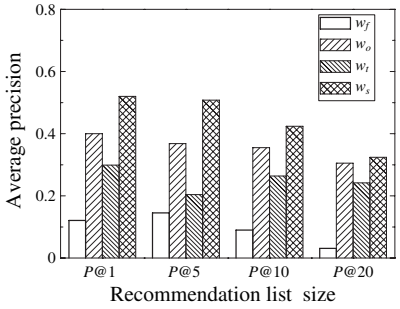


**Figure 4** Average precision for different social features.

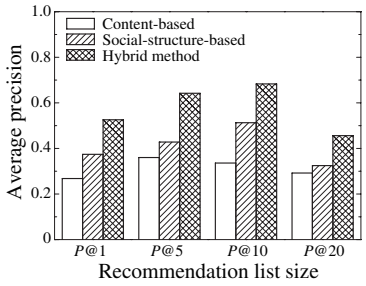


**Figure 5** Average precision for different factors.

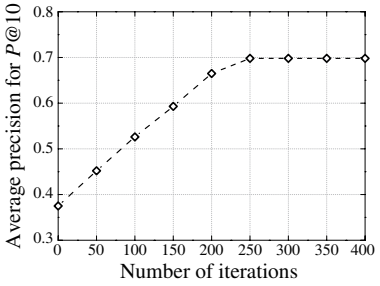


**Figure 6** Simulated Annealing for $P$@10.

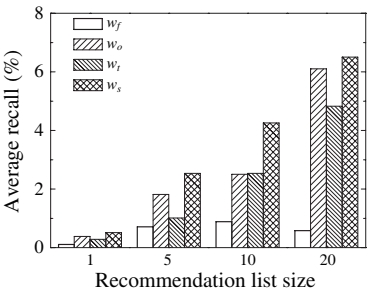


**Figure 7** Average recall for different social features.

### 5.3 Results

Figure 4 shows the comparison of different social structure factors, $w_f(u)$, $w_o(u)$ and $w_t(u)$ as described in (10). The results show that ranking solely with $w_o(u)$, the factor considering the number of occurrences of the candidate user in the final list of $R$, has better precision scores than any of the other two factors exploited. We obtain a good recommendation in the first position of the ranking for 40.5% of the users. For longer top-k lists, precisions decrease from 37.1% at $P$@5 to 30.6% at $P$@20. In contrast, ranking with the factor $w_f(u)$ considering the ratio of one's followers and followees get the worst precision of less than 15% in different top-k lists. A middle influence factor is $w_t(u)$, the weighting feature considering the number of friends shared between the target user and the candidate user. The precision varies from 30.2% at $P$@1 to 20.5% at $P$@5.

We examine the combination of all the above social factors as a whole using $W_s(u)$ presented in (10).

The results show that when compared with the ranking scheme exploiting any single factor, the ranking with the optimized combination of those social factors achieves a much better precision across all the target users, with the highest precision of 52.5% at $P$@1 and the lowest precision of 32.6% at $P$@20.

Figure 5 shows that simply ranking with content relevance information $W_c$ has slightly lower precision compared to the ranking scheme considering social factor $W_s$ combinations. This reveals that social structure information is a little more important than the content factor for followee recommendation in Sina Weibo microblogging system. The slight difference of the importance in Figure 5 also reflects that the content relevance is not trivial and should not be ignored.

We further examine the hybridization of the content factor and the social factor. The results in Figure 5

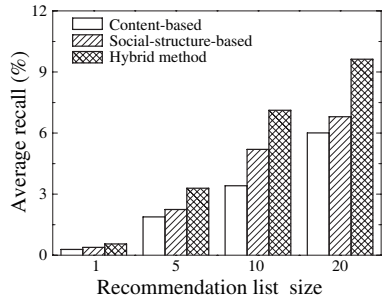


**Figure 8** Average recall for different factors.

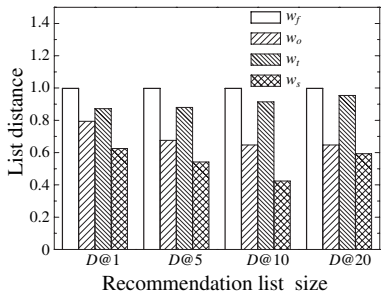


**Figure 9** List distance for different social features.

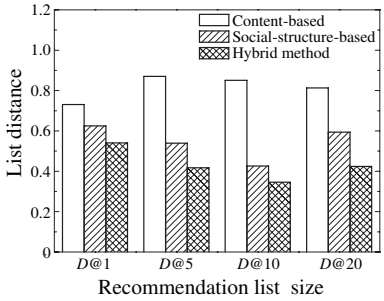


**Figure 10** List distance for different factors.

shows that, by combining the information sources of both content relevance $W_c$ and the social structure $W_s$, our scheme greatly outperforms existing schemes solely using content relevance or social structure. The results show that our scheme improves the precision of the top-10 recommendation by 109% and 37% compared to the recommendation based on content relevance and that based on social structure, respectively.

Figure 6 shows the performance of our SAWA algorithm. The parameter assignment differs from each other according to the size of recommendation list. Recommending 10 followees for the target user, for instance, we find that the algorithm tries around 300 iterations to find out the best assignment. We further try another 1000 iterations, however, find no further improvement. This result reveals that our SAWA algorithm can achieve the improvement of recommend result without suffering from time-consuming problem.

Figure 7 illustrates the results of recall using the algorithms considering different social structure features. It shows that almost all features have increasing tendency with the enlargement of recommendation list size. Using $w_f(u)$ continuously suffers a lower recall when the recommendation list size changes, since it recommends popular followees for all the users while neglecting their personalized attributes. We still use $w_s(u)$ as the combination of all the above social structure factors and exam its performance. The result shows that compared with single factors, the recall of $w_s(u)$ also achieve higher ratios.

Figure 8 shows that structure information has a higher recall than content factor, while the hybrid method combining both of the two factors further outperforms solely using content relevance or social structure. When the size of recommendation list is 10, for example, our metric improves the recall by 108% and 36% compared to the recommendation based on content relevance and that based on social structure, respectively.

Figure 9 illustrates the list distance using the algorithms considering different social structure features. The result shows that the algorithm considering only the factor $w_f(u)$ suffers the largest list distance across all the positions which vary from 1.0 at $D$@1 to 0.995 at $D$@20. The algorithm considering the factor $w_o(u)$ performs a little better, with the list distance ranging from 0.794 at $D$@1 to 0.646 at $D$@20. The results of the algorithms considering $w_t(u)$ and $w_f(u)$ are similar, both showing a large list distance. The result in Figure 8 also shows that by comprehensively considering all the three features, the algorithm using $w_s(u)$ has much better performance than the algorithm considering just one feature.

Figure 10 shows the ranking algorithm which considers the content relevance information has higher list distance compared to the ranking algorithm considering social structure information. We further examine the combination of the content relevance and the social factors. The results in Figure 10 show that, by considering the information sources of both content relevance $W_c$ and the social structure $W_s$, our scheme greatly outperforms existing schemes simply using content factor or social structure in terms of list distance. The results show that our scheme decrease the list distance of the top-10 recommendation by 60% and 19% compared to the recommendation based on content relevance and the recommendation based on social structure, respectively.

Similar to Figure 6, Figure 11 plots the performance of our SAWA algorithm when we recommend 10 followees for each target user. The algorithm tries around 250 iterations to figure out the optimal

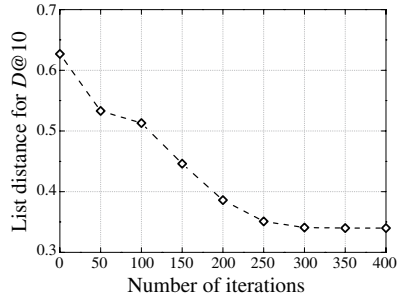


**Figure 11** Simulated Annealing for $D@10$.

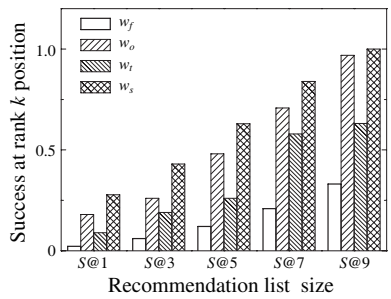


**Figure 12** $S@k$ for different weighting features.

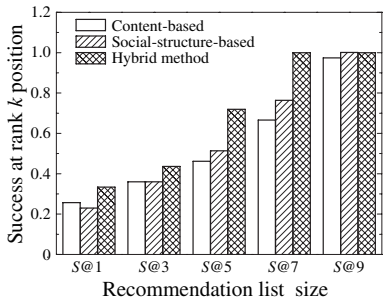


**Figure 13** $S@k$ for different factors.

assignment. When we try another 1000 iterations, we get no further improvement. Again, this result reveals that our SAWA algorithm is effective and efficient.

We further examine the metric success at top-k ($S@k$), a widely accepted metric in this area to evaluate the ranked lists of recommendations [21]. The metric $S@k$ is defined as the probability of finding a good recommendation among the top-k recommended users, i.e., the percentage of runs of tests which locate at least one relevant user among the first $k$ recommended users. Since a user always only pays attention to the top ranks of the recommendation results (especially when he/she uses a mobile client), once the recommendation list size is fixed, the higher value of $S@k$ indicates that the algorithm has a better ability to recommend relevant results. Figure 12 shows the results with values of $k$ ranging from one to nine. Results show that ranking users with $w_o(u)$ has better performance than those with social features $w_f(u)$ and $w_t(u)$, while the combination of all the social features, i.e., $W_s(u)$, yields the best performance. By using $W_s(u)$ the probability that the top one user recommended is relevant is improved by the factor of 0.6–13 compared to the ranking scheme using a single social structure feature. When the size of recommendation list increases, the $S@k$ will all be equal to one.

Figure 13 shows the ranking algorithm which considers the content relevance information has a slightly lower $S@k$ value compared with the ranking algorithm considering social structure information. The results show that our hybrid scheme greatly outperforms either of the schemes based on a single source of information. Using our scheme, the values of $S@k$ ($k \geqslant 7$) hit 100%, indicating that our design is quite promising for the microblogging system.

We further compare the performance of our scheme with two existing popular schemes widely used in the IR field, including the PageRank and LDA scheme [42]. We also compare our scheme with Twittomender [6]. In the following, we first describe the baseline schemes we will further compare and then present the results.

• PageRank: We apply PageRank to the microblogging network of followees and followers. In the network a node maps to a user, and a directed edge maps to following or followed relationships among users. It is clear that the heuristic algorithm tends to recommend popular users to the target user.

• LDA: LDA [42] is a generative topic model and each document is viewed as a mixture of topics. We regard the tweets of users as documents here and learn an LDA model from tweets. After getting the topic distribution of each user, given a target user $U_T$ and a certain candidate user $U_R$, the relevance score is calculated as below:

$$y_{U,T} = \sum_{U_0} \alpha^{I(U_T,U_0)} D_{\mathrm{KL}}(U_0||U_R). \tag{14}$$

Here $D_{\mathrm{KL}}(U_0||U_R)$ calculates the symmetric KL-divergence between the topic distribution of two tweets, the indicator function $I(.)$ equals to one if the user has followed $U_0$ and equals to zero otherwise.

• Twittomender: Twittomender is designed by Hannon et al. [6], as a recommendation system. The system has two major strategy options, indexing users with their content of tweets and indexing users with their neighbor's ID, called as tweet-based approach and ID-based approach, respectively. We use the method with the better performance as the baseline scheme.

Figure 14 compares the average precisions of different schemes. The PageRank suffers the lowest

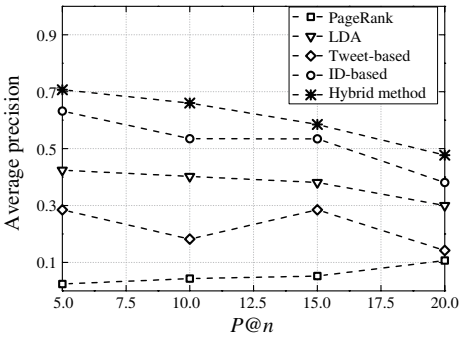


**Figure 14** Average precision.

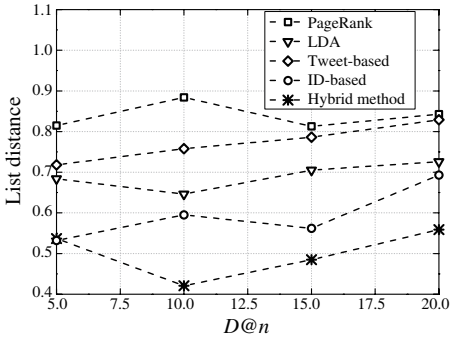


**Figure 15** List distance.

precision since it recommends the same most popular followees to each user which neglects individual difference. The ID-based scheme has a higher precision than both tweet-based scheme and the LDA scheme. The results in Figure 15 also show that our scheme outperforms other schemes. It improves the precision of the top-10 recommendation by 305% and 16% compared to the tweet-based scheme and ID-based scheme, respectively.

Figure 15 shows PageRank has the highest list distance which means only considering the popularity of potential followees is not enough to make good recommendations. LDA topic model performs a little worse than ID-based scheme, for example, 9% at $D$@10. The results show that social structure information is more important than content factor when making followee recommendation, meanwhile, their small difference also indicates that content factor is non-trivial and essential information source. The performance of our approach improves the list distance by 81% and 30% with the top-10 results compared to the tweets-based scheme and ID-based scheme, respectively.

## 6 Conclusion and future work

In this paper we present a novel hybrid scheme for followee recommending in the microblogging system. Our design is based on the observation that microblogging systems have dual roles of social network and news media platform. Accordingly, in our scheme both the information sources of content relevance and social structure are considered. Specifically, we propose a linear weighted model to combine the two factors. To fast approach the optimal settings of the parameters in the model, we further design the SAWA algorithm, which adapts the simulated annealing algorithm. The SAWA algorithm automatically assigns the weights of factors in the linear weighted model. We evaluate this design using large-scale of real-world datasets collected from Sina Weibo. Experiment results show that this hybrid design greatly outperforms existing schemes in terms of recommendation accuracy.

Our scheme can make good followee recommendation for users whose tweets contextual information and social link information are sufficient. Our future research will target at how to make reasonable followee recommendation for newly registered users whose relevant information is not enough to analyze the target user's interest in our present scheme.

**Acknowledgements** This work was supported in part by National Natural Science Foundation of China (Grant Nos. 61370233, 61422202), Research Fund of Guangdong Province (Grant No. 2015B010131001), and Foundation for the Author of National Excellent Doctoral Dissertation of China (Grant No. 201345).

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1 Chen H C, Chen A L P. A music recommendation system based on music data grouping and user interests. In: Proceedings of ACM Conference on Information and Knowledge Management (CIKM), Atlanta, 2001. 231–238

2 Devi M K K, Venkatesh P. Smoothing approach to alleviate the meager rating problem in collaborative recommender systems. Future Gener Comput Syst, 2013, 29: 262–270

3 Guan Z, Wang C, Bu J, et al. Document recommendation in social tagging services. In: Proceedings of International Conference on World Wide Web (WWW), Raleigh, 2010. 391–400
4 Tserpes K, Aisopos F, Kyriazis D, et al. A recommender mechanism for service selection in service-oriented environments. Future Gener Comput Syst, 2012, 28: 1285–1294
5 Breese J S, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of International Conference on Uncertainty in Artificial Intelligence, Madison, 1998. 43–52
6 Hannon J, Bennett M, Smyth B. Recommending twitter users to follow using content and collaborative filtering approaches. In: Proceedings of ACM Conference on Recommender Systems, Barcelona, 2010. 199–206
7 Anagnostopoulos A, Becchetti L, Castillo C, et al. Online team formation in social networks. In: Proceedings of International Conference on World Wide Web (WWW), Lyon, 2012. 839–848
8 Tang J, Sun J, Wang C, et al. Social influence analysis in large-scale networks. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Paris, 2009. 807–816
9 Zhang Z. Community structure detection in social networks based on dictionary learning. Sci China Inf Sci, 2013, 56: 078103
10 Armentano M G, Godoy D, Amandi A. Followee recommendation based on text analysis of micro-blogging activity. Inform Syst, 2013, 38: 1116–1127
11 Kwak H, Lee C, Park H, et al. What is twitter, a social network or a news media? In: Proceedings of International Conference on World Wide Web (WWW), Raleigh, 2010. 591–600
12 Konstan J A, Miller B N, Maltz D, et al. Grouplens: applying collaborative filtering to usenet news. Commun ACM, 1997, 40: 77–87
13 Balabanovi M, Shoham Y. Fab: content-based, collaborative recommendation. Commun ACM, 1997, 40: 66–72
14 Kautz H, Selman B, Shah M. Referral web: combining social networks and collaborative filtering. Commun ACM, 1997, 40: 63–65
15 Rashid A M, Lam S K, Karypis G, et al. Clustknn: a highly scalable hybrid model-and memory-based cf algorithm. In: Proceedings of KDD Workshop on Web Mining and Web Usage Analysis, Philadelphia, 2006
16 Chan S, Jin Q. Collaboratively shared information retrieval model for e-learning. In: Proceedings of International Conference on Advances in Web based Learning, Penang, 2006. 259–266
17 Pazzani M, Muramatsu J, Billsus D. Syskill and webert: identifying interesting web sites. In: Proceedings of National Conference on Artificial Intelligence, Portland, 1996. 54–61
18 Mooney R J, Roy L. Content-based book recommending using learning for text categorization. In: Proceedings of ACM Conference on Digital Libraries, San Antonio, 2000. 195–204
19 Kywe S M, Lim E P, Zhu F. A survey of recommender systems in twitter. In: Proceedings of International Conference Social Informatics, Lausanne, 2012. 420–433
20 Easley D, Kleinberg J. Networks, Crowds, and Markets. Cambridge: Cambridge University Press, 2010
21 Armentano M G, Godoy D, Amandi A. A topology-based approach for followees recommendation in twitter. In: Proceedings of IJCAI Workshop on Intelligent Techniques for Web Persinalization and Recommender Systems, Barcelona, 2011. 144–153
22 Chechev M, Georgiev P. A multi-view content-based user recommendation scheme for following users in twitter. In: Proceedings of International Conference Social Informatics, Lausanne, 2012. 434–447
23 Kywe S M, Hoang T A, Lim E P, et al. On recommending hashtags in twitter networks. In: Proceedings of International Conference Social Informatics, Lausanne, 2012. 337–350
24 Yen N, Shih T, Jin Q, et al. Automatic learning sequence template generation for educational reuse. In: Proceedings of IEEE International Conference on Granular Computing, Kaohsiung, 2011. 259–266
25 Hill W, Terveen L. Using frequency-of-mention in public conversations for social filtering. In: Proceedings of ACM Conference on Computer Supported Cooperative Work (CSCW), Boston, 1996. 106–112
26 Andersen R, Borgs C, Chayes J, et al. Trust-based recommendation systems: an axiomatic approach. In: Proceedings of International Conference on World Wide Web (WWW), Bejing, 2008. 199–208
27 Shardanand U, Maes P. Social information filtering: algorithms for automating word of mouth. In: Proceedings of SIGCHI Conference on Human Factors in Computing Systems, Denver, 1995. 210–217
28 Chen J, Nairn R, Nelson L, et al. Short and tweet: experiments on recommending content from information streams. In: Proceedings of International Conference on Human Factors in Computing Systems (CHI), Atlanta, 2010. 1185–1194
29 Armentano M G, Godoy D, Amandi A. Topology-based recommendation of users in micro-blogging communities. J Comput Sci Technol, 2012, 27: 624–634
30 Golder S A, Yardi S, Marwick A, et al. A structural approach to contact recommendations in online social networks. In: Proceedings of Workshop on Search in Social Media, Boston, 2009
31 Assent I. Actively building private recommender networks for evolving reliable relationships. In: Proceedings of IEEE International Conference on Data Engineering (ICDE), Shanghai, 2009. 1611–1614
32 Xu T, Chen Y, Jiao L, et al. Scaling microblogging services with divergent traffic demands. In: Proceedings of International Middleware Conference (Middleware), Lisbon, 2011. 20–40
33 Naveed N, Gottron T, Kunegis J, et al. Bad news travel fast: a content-based analysis of interestingness on twitter. In: Proceedings of ACM Conference in WebSci, Koblenz, 2011. 1–7
34 Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. Inform Process Manag, 1988, 24: 513–523
35 Garcia R, Amatriain X. Weighted content based methods for recommending connections in online social networks. In: Proceedings of Workshop on Recommender Systems and the Social Web, Hong Kong, 2010. 68–71

36 Kirkpatrick S, Jr D G, Vecchi M P. Optimization by simmulated annealing. Science, 1983, 220: 671–680
37 Nie Z, Zhang Y, Wen J R, et al. Object-level ranking: bringing order to web objects. In: Proceedings of International World Wide Web Conference (WWW), Chiba, 2005. 567–574
38 Wen X, Shao L, Fang W, et al. Efficient feature selection and classification for vehicle detection. IEEE Trans Circ Syst Video Technol, 2015, 25: 508–517
39 Zhang H, Wu Q J, Nguyen T M, et al. Synthetic aperture radar image segmentation by modified student's t-mixture model. IEEE Trans Geosci Rem Sens, 2014, 52: 4391–4403
40 Gu B, Sheng V S. Feasibility and finite convergence analysis for accurate on-line-support vector machine. IEEE Trans Neural Netw Learn Syst, 2013, 24: 1304–1315
41 Li J, Li X, Yang B, et al. Segmentation-based image copy-move forgery detection scheme. IEEE Trans Inform Forens Secur, 2015, 10: 507–518
42 Blei D, Jordan M A Y. Latent dirichlet allocation. J Mach Learn Res, 2003, 40: 993–1022
43 Wu K, Xiao J, Yi Y, et al. CSI-based indoor localization. IEEE Trans Parall Distrib Syst, 2013, 24: 1300–1309
44 Li H, Wu K, Zhang Q, et al. CUTS: improving channel utilization in both time and spatial domains in wlan. IEEE Trans Parall Distrib Syst, 2014, 25: 1413–1423
45 Voorhees E M. Overview of trec 2003. In: Proceedings of Text Retrieval Conference (TREC), Gaithersburg, 2003. 1–13
46 Fu Z, Sun X, Liu Q, et al. Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing. IEICE Trans Commun, 2015, 98: 190–200