

Who Should Share What? Item-level Social Influence Prediction for Users and Posts Ranking

Peng Cui
Computer Science
Department
Tsinghua University
Beijing, China
cuip@tsinghua.edu.cn

Fei Wang
Healthcare Transformation
Group
IBM T J Watson Research
Center
Hawthorne, USA
fwang@us.ibm.com

Shaowei Liu
Computer Science
Department
Tsinghua University
Beijing, China
liushaowei@sina.com

Mingdong Ou
Computer Science
Department
Tsinghua University
Beijing, China
oumingdong@gmail.com

Shiqiang Yang
Computer Science
Department
Tsinghua University
Beijing, China
yangshq@tsinghua.edu.cn

Lifeng Sun
Computer Science
Department
Tsinghua University
Beijing, China
sunlf@tsinghua.edu.cn

ABSTRACT

People and information are two core dimensions in a social network. People sharing information (such as blogs, news, albums, etc.) is the basic behavior. In this paper, we focus on predicting *item-level* social influence to answer the question *Who should share What*, which can be extended into two information retrieval scenarios: (1) Users ranking: given an item, who should share it so that its diffusion range can be maximized in a social network; (2) Web posts ranking: given a user, what should she share to maximize her influence among her friends. We formulate the social influence prediction problem as the estimation of a user-post matrix, in which each entry represents the strength of influence of a user given a web post. We propose a *Hybrid Factor Non-Negative Matrix Factorization* (HF-NMF) approach for item-level social influence modeling, and devise an efficient projected gradient method to solve the HF-NMF problem. Intensive experiments are conducted and demonstrate the advantages and characteristics of the proposed method.

Categories and Subject Descriptors

J.4 [Social and Behavioral Sciences]: Sociology; H.1.2 [Information Systems]: Models and Principles—*Human factors*

General Terms

Algorithms

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGIR '11, July 24–28, 2011, Beijing, China.

Copyright 2011 ACM 978-1-4503-0757-4/11/07 ...\$10.00.

Keywords

social influence; matrix factorization; user ranking; web post ranking

1. INTRODUCTION

With the rapid proliferation of social applications, such as Flickr, Facebook, Twitter, etc., more and more user profiles, interactions, and collective intelligence (such as social tags, comments, etc.) are available online, which opens a new perspective for information retrieval applications where more focus should be put on user collaborative information. At the same time, new search scenarios, such as web people search [12] and relationship search [10], also emerged. These scenarios propose a new challenge to traditional information retrieval: *how to effectively handle the social information?*

One key concept related to this challenge is *Social influence*, which has been becoming a prevalent and complex force governing the dynamics of people relationships or social network [20]. It is also a key dimension for modern information retrieval in multiple aspects. To mention a few, (1)each user act as an information source in social network, and the influence of a user is meaningful for the authority of the generated information; (2)in web people search, the social influence is the key indicator for influencer search; (3)as different information varied with the power to affect people to change their actions, they can be recommended by influence ranking for social purpose. Therefore, there is a clear need for techniques to analyze social influence, and more importantly, in information retrieval field. However, to the best of our knowledge, this issue has still not been well studied.

The existing social influence analysis research can be summarized into a diagram: *Who(A) influences Whom(B) given What(C)*. A is often regarded as a single user (or node). For B, previous works can be categorized as *macroscale*, where B is the whole network [2, 20]; *microscale*, where B is a single user [7, 18]; and *mesoscale*, where B is the community or A's friends (neighborhoods) [13, 1]. From the side of C, there are also three lines of research: Structure-level, when

C is empty [14, 17]; Topic-level when C is a topic [18, 8]; and Item-level when C is an item (such as a web page, product, etc.), which is rare in previous research.

In this paper, we focus on measuring item-level social influence in mesoscale to answer the question *Who should share What*, which can be extended into two information retrieval scenarios: (1) Users Ranking: given an item (C), who (A) should share it so that its diffusion range can be maximized in a social network; (2) Web Posts Ranking: given a user (A), what (C) should she share to maximize her influence among her neighbors.

1.1 Motivating Application

In social computing, people and information are two core dimensions and people sharing information (such as blog, news, album, etc.) is the basic behavior. Actually, the spreading out of information is because of the user sharing in social network. The owner of the information, e.g. the advertisers, hope to maximize the diffusion range of the information [4]. This goal makes them desire to target the influencers, who are able to let many friends to click the information they share or even share further to extend the sharing cascades. Psychologically, people share information with their friends mainly because they want to build their reputations and help others, in which *to influence others* is the important motivation for sharing [19].

According to the definition of social influence on Wiki, social influence occurs when "an individual's thoughts, feelings or actions are affected by other people". In the context of online social networks like Facebook and Twitter, when a user share a web post, a portion of her friends (or neighbors) will click, comment, or even forward the post, which are three levels of influence [21]. In this paper, we only consider the first level, click action. That is, the social influence of a user on her friends given a web post is defined as the number of her friends who click the shared web post.

Therefore, the aim of this paper is to predict the number of clicks by friends if a user share a web post. The resulted predictive model can be used in two angles. On one side, given a web post, we can find out the influencers for the diffusion. On the other, given an user, we can recommend a list of web posts to share, which can improve the interactions between the user and her friends.

1.2 Challenges and Contributions

In predicting the item-level social influence, we mainly face the following challenges:

- User-post specific. Item-level social influence is not a general measure on users, but on the interactions of users and posts. That is, we need to discriminate a user's social influences with respect to different web posts. Different from most of the existing research works focusing on users' overall social influence analysis [14, 17] and topical social influence mining [18, 8], the social influence in this paper is a finely grained measure of influence.
- Sparsity. The interactions between users and web posts are extremely sparse compared with the total number of user-post pairs. According to our statistics of 34K users in the website www.renren.com, which is a Facebook style social network site in China, each user only shares 6 web posts in average during a month, com-

pared with a total of 43K web posts; and each post is only shared by 4 users, compared with a total of 34K users. Thus, it is clear that we need subtle and effective prior knowledge for user and post grouping to alleviate the sparsity problem.

- Complex factors. There are a volume of factors that affect how many friends will click a shared post, and provide potential clues for user and post grouping. For example, the total number of friends, the tie strength between the user and her friends, the semantics of web posts, etc., which are often in different scales. How to select the effective factors and integrate these complex factors in one predictive model is also one of the focus of our work.

In this paper, we formulate the social influence prediction problem as the estimation of a user-post matrix, in which each element (i, j) represents the number of clicks by friends of user i on her j -th shared web post. We proposed a *Hybrid Factor Non-negative Matrix Factorization (HF-NMF)* algorithm for item-level social influence modeling. In this model, we try to find out the common hidden vector space for both the users and the posts, where their multiplication can well approximate the observed training interaction matrix. Meanwhile, in order to deal with the sparsity problem, we construct the priors on users and posts by incorporating the user-user similarity matrix and post topic distribution matrix. Also, in order to alleviate the over-fitting problem, we introduce the L2-norm as regulations for the hidden vector space to improve the generalization ability. We apply *Projected Gradient* to solve the HF-NMF problem, and carry out intensive experiments to demonstrate the effectiveness of the proposed method.

It is worthwhile to highlight the key contributions of this paper.

- We formulates the item-level social influence prediction problem formally with HF-NMF, and devise an efficient projected gradient method to solve it.
- The predicted item-level social influence from HF-NMF can support the applications such as influencer ranking and information recommendation by user-post matrix ranking in two directions.
- The strength of social influence in this method is well interpreted, which makes it easy to understand and extendable to higher order social influence, for example, the influence on all the friends and the friends of friends.
- We conducted intensive experiments on real social network datasets, and the results show that the HF-NMF can achieve a better performance compared with other competitors.

The rest of the paper is organized as follows. Section 2 introduces the necessity of item-level social influence and rationality of factor selection. The HF-NMF model and its solution will be proposed in Section 3. The experiment results to validate the proposed method are presented in Section 4, followed by the conclusion in Section 5.

2. PRELIMINARIES

In this section, we will demonstrate the necessity of the item-level social influence, and validate the rationality of predictive factors by preliminary statistical analysis.

2.1 Necessity of Item-level Social Influence

The dataset is acquired from the real social network <http://renren.com/>, which is a Facebook style online social community in China. Till now, the web site already owns more than 150 million active users. In this web site, a user can generate a post or share a web page as a post, and the user’s friends will be informed through the news feed mechanism. Then some of the friends will click, comment or share the post. In this paper, we only consider the click action as the manifestation of influence, and the number of clicks corresponds to the strength of influence. As the number of friends is different for each user, the upper bound of users’ social influence strengths are also different. In order to make the strength of influence be measured in a unified scale for different users for the sake of observational and modeling study, we use the proportion of friends (of the user who publish a post) who click the shared post as the measure.

Concerning about the influence prediction, we first assume that the influence should be *specific* on each user-post pairs. In order to validate the hypothesis, we randomly select three active users. Given an user, we calculate the proportion of her influenced friends (who clicked the shared post) for each of the shared posts, and plot them as in the first row of Figure 1. Then, we randomly select three popular posts. Given a post, we calculate the proportion of influenced friends when the post is shared by different users, and plot them as in the second row of Figure 1.

As can be observed, the social influence notably varies with different users and posts, which implies that (1) different users have different influence power to their friends; (2) different posts have different influence power (more intuitively, attraction) to users who are interested in; and (3) users’ influences manifest differently for different posts. Therefore, only item-level social influence can reveal the users’ real influence on friends, and the strength of influence should definitely be user-post specific.

2.2 Predictive Factors

According to Figure 1, the factors that affect the strength of social influence include the following three aspects.

- **User-specific factors.** Although users’ social influence vary with the shared posts, the average of the social influences over posts determines the overall social influence of a user. We regard the factors that affect users’ overall social influence (excluding the posts) as the user-specific factors.
- **Post-specific factors.** Similar as user-specific factors, we regard the factors affecting posts’ overall social influence (excluding the users) as the post-specific factors.
- **User-post specific factors.** As mentioned above, the social influence is user-post specific. The social influence of a user given a post cannot be well approximates only by the user and post-specific factors. The factors indicating the interactions between users and posts are also important for social influence prediction.

One issue that is worthy of emphasizing here is that the user factors and post factors are essential for the predictive modeling. On one hand, the user-post interactions are very sparse. We need to find effective factors to “group” those users and posts to alleviate the sparsity problem. On the other hand, the user and post-specific factors also provide some effective prior knowledge to complement the inference from pure user-post interactions.

In order to find out the effective predictive factors, we first prepared a factor pool, which includes the available potential predictive factors including user profiles, number of users’ friends, visiting frequency between users, posts’ topic distributions, etc. Given each factor, we measured the correlation between the strength of social influence and the factor value. Finally, we select two user-oriented factors: the percentage of active friends, the average social tie strength (the interaction frequency) between a user and her friends, and one post-specific factor: the topic distribution of a web post’s content.

Now we show some statistics to demonstrate the validity of those selected factors. We randomly select 10 users from the dataset. Given each user, we define her social influence as the average percentage of her friends who click the shared post over all shared posts, and calculate the correlation between the selected user factors and this social influence measure. Figure 2 (a)(b) illustrate the correlation between the user factors and the social influence measure on these 10 users, from which we can clearly observe the positive results.

For post topic distribution, our hypothesis is that the posts with similar contents (i.e. similar topic distributions) often induce similar social influences. To validate this assumption, we randomly select ten groups of web posts, where the posts in the same group have similar topic distributions. Given each web post, we define its social influence as the average percentage of users’ friends who clicked it over all the users who shared the post. Then we calculate the variances of posts’ social influence inside each topic group, and also over all the web posts (i.e. across topic groups). As we can see from Figure 2(c), most variances inside topic groups are smaller than that across the groups, which implies that the introduction of topical grouping is able to reduce down the uncertainty of social influence.

Based on the above analysis, we validate the effective predictive factors and apply them into the predictive modeling in Section 4.

3. THE ALGORITHM

3.1 Problem Formulation

First, we formally define the problem of item-level social influence prediction. Suppose we have M users with the i -th user denoted as u_i and N postings with the j -th post denoted as p_j . We use $\mathcal{N}(u_i)$ to denote the collection of u_i ’s first-order friends (i.e. the nodes that directly link to u_i). As mentioned in the previous section, two key factors involved in our model are

- **Item-level social influence:** According to the previous presentations, a straightforward way to define the strength of u_i ’s influence on $\mathcal{N}(u_i)$ given the web post p_j , denoted as f_{ij} , is the number of u_i ’s friends who clicked post p_j .

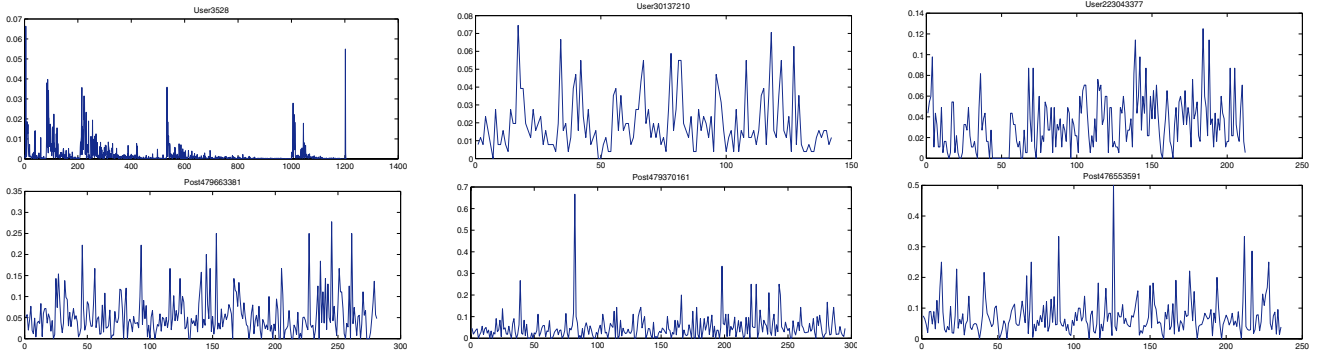
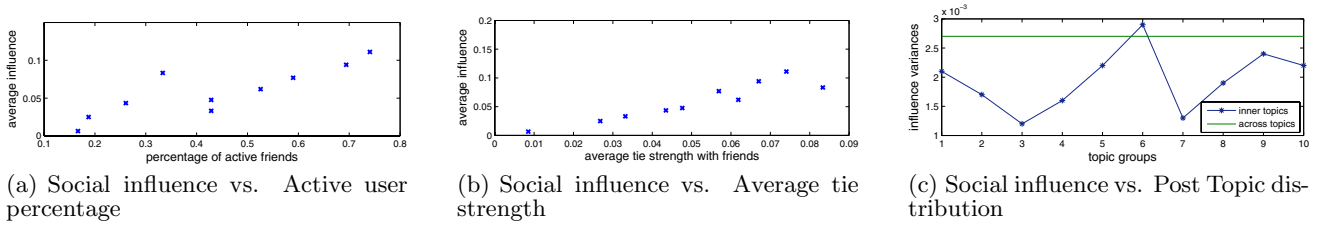


Figure 1: Social influence varies with users and posts.



(a) Social influence vs. Active user percentage

(b) Social influence vs. Average tie strength

(c) Social influence vs. Post Topic distribution

Figure 2: Predictive factors validation.

- **Social influence prediction:** There are $M \times N$ potential social influences in total. However, in practice, only a tiny fraction of them can be observed. The social influence prediction is to predict the unobserved social influences \hat{f}_{ij} based on the observed f_{ij} 's and those predictive factors.

With the above terminologies, we can formally define the task of item-level social influence prediction. We denote the user-post influence matrix as $\tilde{\mathbf{X}} \in \mathbb{R}^{M \times N}$, with its (i, j) -th entry

$$\tilde{X}_{ij} = \begin{cases} f_{ij} & \text{if } u_i \text{ shared } p_j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

If we use g_i to denote the number of u_i 's friends (i.e. $g_i = |\mathcal{N}(u_i)|$, where $|\cdot|$ is the cardinality of a collection), then $f_{ij} \leq g_i$. Also, It should be noted that in the matrix $\tilde{\mathbf{X}}$, there are two cases where an entry \tilde{X}_{ij} is 0. First is that u_i did not share the p_j , and the second is that u_i shared p_j , but no friends of u_i clicked it.

As we mentioned above, different users have different numbers of friends, which makes the strength of social influence (if measured by f_{ij}) for each user-post pair be measured in different scales. To alleviate its effect on the final performance, we propose the following *percentile* influence matrix

$$X_{ij} = \begin{cases} \frac{f_{ij}}{g_i} & \text{if } u_i \text{ shared } p_j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

so that X_{ij} 's are normalized into the range of $[0, 1]$.

The user-post influence matrix $\tilde{\mathbf{X}}$ can be reconstructed by

$$\tilde{\mathbf{X}} = \text{Diag}(\mathbf{g}) \cdot \mathbf{X} \quad (3)$$

where $\mathbf{g} = [g_1, g_2, \dots, g_N]^\top \in \mathbb{R}^N$, and $\text{Diag}(\mathbf{g})$ is the diagonal matrix with \mathbf{g} on the diagonal line.

In this way, the item-level social influence prediction problem is converted to the problem of predicting the unobserved entries in \mathbf{X} .

Our formulation of the item-level social influence prediction problem is quite different from existing works on social network analysis. First, we measure the social influence in item-level, compared with the structure-level analysis [14, 17] and topic-level analysis works [18, 8]. Second, the goal of the problem is to predict the users' social influence for unobserved data, which is in contrast with the majority of existing works to analyze the influence patterns from observed data [2, 3].

In the following section, we will formally present the *Hybrid Factor Nonnegative Matrix Factorization* (HF-NMF) approach to model the problem.

3.2 Hybrid Factor Non-negative Matrix Factorization

We suppose that there exists a joint latent space for both users and posts with dimensionality k , such that the user-post specific social influences are modeled as the inner product between user-post vector pairs in that space. Accordingly, the user u_i is associated with an user vector $\mathbf{u}_i \in \mathbb{R}^k$, and the post p_j is associated with a post vector $\mathbf{v}_j \in \mathbb{R}^k$. Intuitively, the elements in \mathbf{v}_j measures the extent to which the post p_j possess those k factors, and the elements of \mathbf{u}_i measure the extent of social influence the user i has in posts that are high on the corresponding factors.

Now we will formulate the social influence prediction problem mathematically. Suppose that we have M users and N web posts. Let $\mathbf{U} \in \mathbb{R}^{M \times k}$ be the latent user feature matrix, and $\mathbf{V} \in \mathbb{R}^{N \times k}$ be the latent poster feature matrix, where k is the number of latent features. Then given the observed user-post specific social influence matrix \mathbf{X} , the objective of

this paper is to find the optimal latent user matrix \mathbf{U} and latent post matrix \mathbf{V} by minimizing the following objective.

$$\mathcal{J}_1 = \left\| \mathbf{X} - \mathbf{U}\mathbf{V}^\top \right\|_F^2 \quad (4)$$

where $\|\cdot\|_F$ denotes the matrix Frobenius Norm.

The objective function \mathcal{J}_1 can be regarded as the quality of approximating \mathbf{X} by the inner product of \mathbf{U} and \mathbf{V} . However, in real cases, most of the elements in \mathbf{X} are zero because of the sparse interactions between users and posts. Thus, in order to focus more on the valid elements, we propose to only measure the approximation loss on observed elements on \mathbf{X} . To formulate this, we introduce the *sharing matrix* $\mathbf{Y} \in \mathbb{R}^{M \times N}$ with its (i, j) -th entry defined as

$$Y_{ij} = \begin{cases} 1 & \text{if } u_i \text{ shared } p_j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The objective function is converted to

$$\mathcal{J}_2 = \left\| \mathbf{Y} \odot (\mathbf{X} - \mathbf{U}\mathbf{V}^\top) \right\|_F^2 \quad (6)$$

where \odot is the Hadamard product.

As mentioned above, the severe sparsity of \mathbf{X} makes it very challenging to directly learn the latent spaces for users and posts from only observed user-post interaction entries. That's the reason why we need to make full use of the user-specific and post-specific factors to compress the degrees of freedom, so that the correlation within users and web posts can be exploited to alleviate the sparsity problem.

3.2.1 Incorporating User-Specific Factors

As introduced in Section 2, we explored two effective user-oriented predictive factors: the *percentage of active friends*, and the *average friend tie strength*.

In order to calculate the percentage of active friends, we define the activeness of a friend u_r , denoted by $act(u_r)$, as the number of posts she visits during a given time period, so that the percentage of active friends for u_i is calculated by

$$uf_1(u_i) = \frac{\sum_{u_r \in \mathcal{N}(u_i)} \delta(act(u_r) \geq \tau)}{|\mathcal{N}(u_i)|} \quad (7)$$

where τ is the time threshold for active user, and $\delta(\cdot)$ is the Delta function.

Also, we define the *tie strength* between a user u_i and one of her friends u_r as the number of shared posts (by user u_i) friend u_r visits, which is denoted as $tie(u_i, u_r)$. Then, the average friend tie strength is calculated by

$$uf_2(u_i) = \frac{\sum_{u_r \in \mathcal{N}(u_i)} \frac{tie(u_i, u_r)}{\sum_j Y_{ij}}}{|\mathcal{N}(u_i)|}. \quad (8)$$

Then we use these two factors to measure the similarity between u_i and u_j as

$$W_{ij} = \rho_1 |uf_1(u_i) - uf_1(u_j)| + \rho_2 |uf_2(u_i) - uf_2(u_j)| \quad (9)$$

In our case, we set $\rho_1 = \rho_2 = 0.5$.

In this way, we can construct the user-user similarity matrix $\mathbf{W} \in \mathbb{R}^{M \times M}$. We further assume that \mathbf{W} can be approximated by the inner product of the latent user matrix, thus we need to minimize the following objective

$$\mathcal{J}_3 = \left\| \mathbf{W} - \mathbf{U}\mathbf{U}^\top \right\|_F^2 \quad (10)$$

3.2.2 Incorporating Post-Specific Factors

As demonstrated in Section 2, the social influence is strongly correlated with the content of the web posts. We denote the post content matrix as $\mathbf{C} \in \mathbb{R}^{N \times d}$, where d is the dimensionality of the posts, which is constructed by implementing *Latent Dirichlet Allocation* (LDA) [6] on the post corpus to discover 100 topics. Then the content of each post is represented as the topic distributions over the 100 topics. Similar to *latent semantic analysis* [9], we assume \mathbf{C} can be factorized by minimizing

$$\mathcal{J}_4 = \left\| \mathbf{C} - \mathbf{V}\mathbf{G}^\top \right\|_F^2 \quad (11)$$

where the matrix $\mathbf{G} \in \mathbb{R}^{k \times M}$ indicates post group identity.

Finally, by combining $\mathcal{J}_2, \mathcal{J}_3, \mathcal{J}_4$ together, we can get the latent user matrix \mathbf{U} and latent post matrix \mathbf{V} by solving the following optimization problem

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}, \mathbf{G}} \quad & \left\| \mathbf{X} - \mathbf{U}\mathbf{V}^\top \right\|_F^2 + \alpha \left\| \mathbf{W} - \mathbf{U}\mathbf{U}^\top \right\|_F^2 + \beta \left\| \mathbf{C} - \mathbf{V}\mathbf{G}^\top \right\|_F^2 \\ & + \gamma \left\| \mathbf{U} \right\|_F^2 + \delta \left\| \mathbf{V} \right\|_F^2 \\ \text{s.t.} \quad & \mathbf{U} \geq 0, \mathbf{V} \geq 0, \mathbf{G} \geq 0 \end{aligned} \quad (12)$$

The last two terms of the objective are penalized for generalization purpose, which is similar to the *Maximum Margin Matrix Factorization* approach in [15] and the *Joint Matrix Factorization* approach in [?].

3.3 Solution

In this section, we will introduce a *Projected Gradient* (PG) method [11] to solve problem (12).

Before going into the details, we first introduce the basic PG algorithm [11]. For notational convenience, we introduce a nonnegativity projection operator $P[\cdot]$ for matrix \mathbf{A} as

$$(P[\mathbf{A}])_{ij} = \begin{cases} A_{ij} & \text{if } A_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

Then the PG method for solving the problem

$$\min_{\mathbf{A} \geq 0} f(\mathbf{A}) \quad (14)$$

can be presented in Algorithm 1.

Algorithm 1 Projected Gradient

Require: $0 < \beta < 1, 0 < \sigma < 1$. Initialization $\mathbf{A}^{(0)}$.

Ensure: $\mathbf{A}^{(0)} \geq 0$

for $k = 1, 2, \dots$ **do**

$$\mathbf{A}^{(k)} = P \left[\mathbf{A}^{(k-1)} - \alpha_k \nabla f \left(\mathbf{A}^{(k-1)} \right) \right]$$

where $\alpha_k = \beta^{t_k}$, and t_k is the first nonnegative integer for which

$$f(\mathbf{A}^{(k)}) - f(\mathbf{A}^{(k-1)}) \leq \sigma \left\langle \nabla f(\mathbf{A}^{(k-1)}), (\mathbf{A}^{(k)} - \mathbf{A}^{(k-1)}) \right\rangle \quad (15)$$

end for

In Algorithm 1, $\nabla f(\mathbf{A})$ is the gradient of the objective function with respect to the variable \mathbf{A} , $\langle \cdot, \cdot \rangle$ is the sum of the component-wise product of two matrices.

Here condition (15) ensures the sufficient decrease of the function value per iteration, and this rule of determining the stepsize is usually referred to as the *Amijo rule* [5]. However,

the Amijo rule is usually time consuming, thus we use the following improved PG method in Algorithm 2 [11].

Algorithm 2 Improved Projected Gradient

Require: $0 < \beta < 1, 0 < \sigma < 1$. Initialization $\mathbf{A}^{(0)}, \alpha_0 = 1$.

Ensure: $\mathbf{A}^{(0)} \geq 0$

for $k = 1, 2, \dots$ **do**

 Assign $\alpha_k = \alpha_{k-1}$

 If α_k satisfies condition (15), repeatedly increase it by

$$\alpha_k \leftarrow \alpha_k / \beta$$

 until either α_k does not satisfy (15) or $\mathbf{A}(\alpha_k / \beta) = \mathbf{A}(\alpha_k)$

 Else repeatedly decrease α_k by

$$\alpha_k \leftarrow \alpha_k \cdot \beta$$

 until α_k satisfies condition (15).

 Set $\mathbf{A}^{(k)} = P \left[\mathbf{A}^{(k-1)} - \alpha_k \nabla f \left(\mathbf{A}^{(k-1)} \right) \right]$.

end for

Now let's return to problem (12). Although it is not jointly convex with respect to \mathbf{U}, \mathbf{V} and \mathbf{G} , it is convex with each of them with the other two fixed. Therefore we can adopt a *block coordinate descent* scheme to solve the problem [5]. That is, starting from some random initialization on $\mathbf{U}, \mathbf{V}, \mathbf{G}$, we solve each of them alternatively with the other two fixed, and proceed step by step until convergence¹. Specifically, the gradients of the objective (denoted as \mathcal{J}) with respect to the variables are

$$\frac{\partial \mathcal{J}}{\partial \mathbf{U}} = 2 \left(-(\mathbf{Y} \odot \mathbf{X})\mathbf{V} + (\mathbf{Y} \odot \mathbf{U}\mathbf{V}^\top)\mathbf{V} - 2\alpha\mathbf{W}\mathbf{U} + 2\alpha\mathbf{U}\mathbf{U}^\top\mathbf{U} + \gamma\mathbf{U} \right)$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{V}} = 2 \left(-(\mathbf{Y}^\top \odot \mathbf{X}^\top)\mathbf{U} + (\mathbf{Y}^\top \odot \mathbf{V}\mathbf{U}^\top)\mathbf{U} - \beta\mathbf{C}\mathbf{G} + \beta\mathbf{V}\mathbf{G}^\top\mathbf{G} + \delta\mathbf{V} \right)$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{G}} = 2\beta \left(-\mathbf{C}^\top\mathbf{V} + \mathbf{G}\mathbf{V}^\top\mathbf{V} \right)$$

4. EXPERIMENTS

In this section, we will introduce the experimental results on a real world dataset to demonstrate the effectiveness of the proposed method. First we will introduce the dataset information.

4.1 Experimental Dataset

We perform our experiments on a real online social network dataset, which is crawled from <http://renren.com>, a Facebook style social network web site in China. We have 34k users, and 43k web posts in the dataset, and the basic information we used for each user and post are listed in Table 4.1.

We filter the raw dataset in two aspects: (1) we only use the posts generated at least 6 days before the crawling date,

¹Here the objective is obviously lower bounded by 0, and the alternating PG procedure will decrease it monotonically. Thus the algorithm is guaranteed to be convergent.

user information	post information
user id (UID)	post id
friend links	post content
shared post id list	visiting friend UID list

Table 1: Data set information.

so that the click number of the post is stable; (2) the "dead users" (i.e. users who never shared posts) and "dead posts" (i.e. posts that were never visited) are discarded. After the filtering, we remain 10k users and 10k web posts with 104k user-post sharing interactions for experiments. The sparsity of the dataset (denoted as 10000user dataset) is around 0.1%, which is rather challenging for predictive modeling.

In Figure 3, we show the characteristics of 10000user dataset by plotting (a) histogram of users with respect to the number of shared posts (calculated by $\sum_j Y_{ij}$ for u_i), (b) histogram of posts with respect to the times of being shared (calculated by $\sum_i Y_{ij}$ for p_j), and (c) histogram of percentile item-level social influence (calculated by $\frac{f_{i\cdot}}{g_i}$).

In our experiments, we randomly sample different number of users and select the web posts shared by these sampled users to form datasets with different sizes, including 500users dataset, 2000users dataset, 5000users dataset, and the 10000users dataset. They are used to evaluate the detail performance of the proposed method.

4.2 Comparative Methods

Besides the proposed HF-NMF method, we also implement the following methods for comparison.

- **Logistic Regression (LR):** If we regard the user and post factors as variables, and the strength of social influence as the response, then the prediction of social influence can be formulated as a regression problem. Thus, we firstly use the LR model to linearly combine the user factors and post factors, and learn the regression coefficients of the factors from the observed training data.
- **Cox Proportional Hazards Regression (CoxPH):** Different from LR, the user factors and post factors are combined in an exponential way, as is used in [22], which aims to predict the speed of diffusion of tweets in Twitter.
- **User Averaging Influence (AvgU):** As users have different overall influences regardless of web posts, we can predict unobserved social influence by the average over observed ones, i.e.,

$$f_{i\cdot} = \frac{\sum_j f_{ij}}{\sum_j Y_{ij}}. \quad (16)$$

- **Post Averaging Influence (AvgP):** As in AvgU, we can also predict the social influence by the web posts' averaging influence regardless of users:

$$f_{\cdot j} = \frac{\sum_i f_{ij}}{\sum_i Y_{ij}}. \quad (17)$$

- **Basic Non-Negative Matrix Factorization (bNMF):** In this method, we only consider the user-post inter-

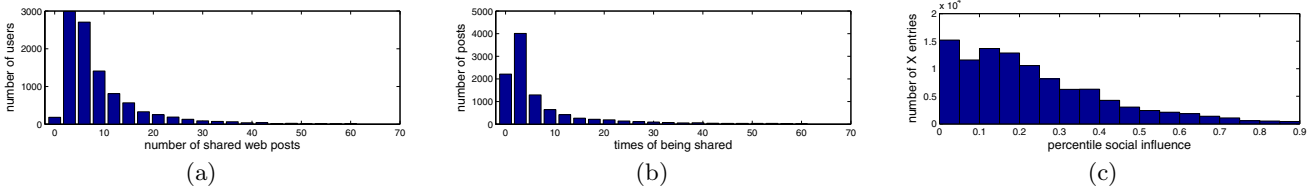


Figure 3: Statistics of the dataset.

action matrix, and find the joint latent space for users and posts by solving the objective function:

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}} \left\| \mathbf{X} - \mathbf{UV}^\top \right\|_F^2 + \gamma \|\mathbf{U}\|_F^2 + \delta \|\mathbf{V}\|_F^2 \\ \text{s.t. } \mathbf{U} \geq 0, \mathbf{V} \geq 0 \end{aligned} \quad (18)$$

- **User Factors Constrained NMF (bNMF+UF):** By incorporating the user factors in to the bNMF, we find the joint latent space for users and posts by solving:

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}} \left\| \mathbf{X} - \mathbf{UV}^\top \right\|_F^2 + \alpha \left\| \mathbf{W} - \mathbf{UU}^\top \right\|_F^2 + \gamma \|\mathbf{U}\|_F^2 + \delta \|\mathbf{V}\|_F^2 \\ \text{s.t. } \mathbf{U} \geq 0, \mathbf{V} \geq 0 \end{aligned}$$

- **Post Factors Constrained NMF (bNMF+PF):** By incorporating the post factors in to the bNMF, we find the joint latent space for users and posts by solving:

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}} \left\| \mathbf{X} - \mathbf{UV}^\top \right\|_F^2 + \beta \left\| \mathbf{C} - \mathbf{VG}^\top \right\|_F^2 + \gamma \|\mathbf{U}\|_F^2 + \delta \|\mathbf{V}\|_F^2 \\ \text{s.t. } \mathbf{U} \geq 0, \mathbf{V} \geq 0, \mathbf{G} \geq 0 \end{aligned}$$

4.3 Evaluation Measures

The main goal of the paper is to learn the joint latent space for users and posts so that their inner-products can approximate the observed entries in \mathbf{X} , and meanwhile, have good prediction performance for unobserved entries. In our experiments, we will hide some observed entries \mathcal{X} by controlling the user sharing matrix \mathbf{Y} in training, and using the learned \mathbf{U} and \mathbf{V} to approximate \mathcal{X} . Then the quality of the prediction will be evaluated using the *Root Mean Square Error* (RMSE).

$$RMSE = \sqrt{\frac{\sum_{X_{ij} \in \mathcal{X}} (X_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)^2}{|\mathcal{X}|}} \quad (19)$$

After predicting the hidden entries, we further rank them along two dimensions (i.e. user dimension and post dimension) according to two application scenarios: influencer ranking and influential post ranking. Here we use two ranking-based evaluation criterions proposed in [16] to measure the ranking performance: (1) T-measure, which simply counts how many of the pairs in the test data are ordered incorrectly, and it is calculated by

$$T = \sum_{i < j} 1(s_i > s_j), \quad (20)$$

where i, j are the orders in groundtruth, and s_i, s_j are the predicted orders. (2) R-measure, which weights the incorrect

α	β	γ	δ	RMSE
0.00001	0.001	0.001	0.001	0.15564
0.0001	0.01	0.01	0.01	0.15234
0.001	0.1	0.1	0.1	0.17742

Table 2: HF-NMF tradeoff parameters setting and evaluation.

ordered pairs by the difference in the testing ranks, and it is calculated by

$$R = \sum_{i < j} (j - i) \cdot 1(s_i > s_j). \quad (21)$$

These two measures are normalized into $[0,1]$ by

$$\begin{aligned} \eta &= 1 - \frac{2T}{n(n-1)} \\ \varrho &= 1 - \frac{6R}{n(n-1)(n+1)} \end{aligned} \quad (22)$$

with 1 corresponding to perfect ranking and 0 representing the reverse ranking.

4.4 Parameter Settings

In this section, we will investigate the effect of different parameter settings when implementing HF-NMF, include tradeoff parameters, dimension of hidden space, and number of projected gradient iterations, on the performance.

4.4.1 Tradeoff Parameters

The tradeoff parameters $\alpha, \beta, \gamma, \delta$ in HF-NMF play the role of adjusting the strength of different terms in the objective function. As the value range of the 4 components in equation (12) are different, the parameter setting should be consistent with the corresponding component's value range. The component $\|\mathbf{W} - \mathbf{UU}^\top\|_F^2$ is the sum of 100 million entries with each entry ranging from 0 to 1. The component $\|\mathbf{C} - \mathbf{VG}^\top\|_F^2$ is the sum of 1 million entries with each entry ranging from 0 to 1. Both $\|\mathbf{U}\|_F^2$ and $\|\mathbf{V}\|_F^2$ are the sum of 10k entries with most entries ranging from 0 to 1. Considering the roles of different components, we test the three sets of tradeoff parameters as shown in Table 2, and use the 2000users dataset for validation.

The results in Table 2 show that the parameter set $\alpha = 0.0001, \beta = \gamma = \delta = 0.01$ produce the best performance. In our following experiments, we just use this parameter setting.

4.4.2 Dimensionality of the Hidden Space

The goal of HF-NMF is to find a k -dimensional joint latent space for users and web posts. How to set k is important for prediction performance. If k is too small, the users and web posts cannot be well represented and discriminated in the latent space. If k is too large, the computational complexity will be greatly increased. Thus, we conduct 5 experiments with k ranging from 5 to 40 on the 2000users dataset. The results are shown in Figure. 4, from which we can see that with the increase on the dimension k , $RMSE$ will reduce gradually. When $k > 30$, the RMSE reduces rather slow. For the concern of the tradeoff between efficiency and prediction precision, we choose $k = 20$ as the latent space dimension in our experiments.

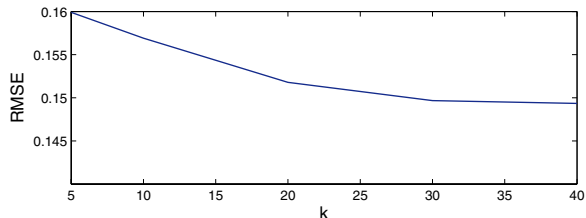


Figure 4: Prediction performance v.s. hidden space dimension.

4.4.3 Number of Iterations

When using PG method to solve problem (12), we need to predefine a proper number of updating iterations to get a good performance while avoid overfitting. Here we use 2000users dataset to validate this parameter. We hide 10% entries from matrix \mathbf{X} and run 50 NMF iterations (each NMF iteration corresponds to update the three matrices \mathbf{U} , \mathbf{V} , \mathbf{G} alternatively with). In each iteration, we record the objective function value and RMSE for hidden entries, and simultaneously plot the convergence curve of PG method and the RMSE curve, as shown in Figure 5.

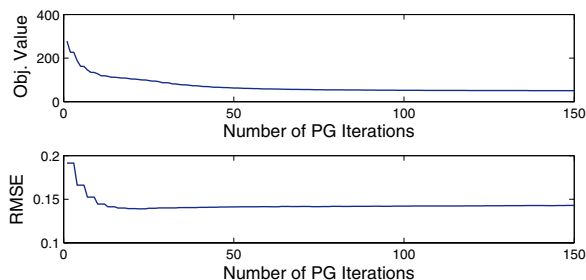


Figure 5: The synchrony between PG convergence curve and prediction performance.

From Figure 5 we can see that the RMSE reduces with the iteration process, as well as the objective function value. Near the 15th PG iteration, the RMSE curve reaches the minimum value. After that, there is a little overfitting effect in RMSE. When the objective function values become stable, the RMSE does not go far away from the minimum value. This demonstrates that the proposed HF-NMF has

good generalization ability. Considering the trade-off between the efficiency and prediction precision, we conduct 45 iterations in PG for the solution in the following experiments.

4.5 Prediction Performance

In this section, we will demonstrate the prediction performance of the proposed method, and compare it with other methods.

We randomly select 50%, 70% and 90% of the observed entries in matrix \mathbf{X} of difference sizes of datasets (including 500users dataset, 2000users dataset, 5000users dataset and 10000users dataset) as the training data, and the rest as the testing data. The random selection was carried out 10 times, and the average RMSE is reported. The same experiments are conducted on the proposed method and the 7 comparative methods listed in subsection 4.2. The results are shown in the Table 3.

From the Table 3, we can observe that:

- The proposed HF-NMF algorithm, which incorporates the user, post and the user-post interaction factors together, achieves the best performance in most experimental trials.
- The more entries used for training, the lower RMSE the methods can achieve. This is consistent with the intuitive assumption that the prediction performance depend heavily on the percentage of training data, especially in sparse dataset where the model can be hardly sufficiently trained.
- HF-NMF's advantage over bNMF is more obvious when there are less observed entries (e.g. 50% than 90%) for training. This is because when the matrix is sparse, the matrix factorization has many possible solutions. In this case, the priors on the users and posts are demanded to compress the degrees of freedom. In HF-NMF, by incorporating the hybrid factors into the matrix factorization model, the sparsity problem is effectively alleviated.
- The comparison between bNMF+UF v.s. bNMF and bNMF+PF v.s. bNMF reveals that the user factors are more effective compared with the post factors in our case.
- The two regression models are not fit for this problem. The main reason is that the regression models can only combine the one side factors (i.e. post-specific factors and user-specific factors), but the interaction between the user and post cannot be well incorporated in the model, which is particularly important in this problem.
- The prediction performance of AvgU is better than AvgP, which reveals that the overall social influence of users excluding web posts are more stable than that of web posts. As both of them do not address the user-post specific variances, their prediction performance are worse than all the matrix factorization methods in all the trails.

4.6 Ranking Task Evaluation

After predicting the social influence, we make use of the results in ranking scenarios. One is to rank the influential

	LR	CoxPH	AvgP	AvgU	bNMF	bNMF+PF	bNMF+UF	HF-NMF
500 users								
50%	0.2026	0.202	0.199376	0.159798	0.158551	0.154719	0.153801	0.152977
70%	0.2093	0.2089	0.185823	0.150239	0.147858	0.149715	0.149009	0.146358
90%	0.1827	0.1822	0.178479	0.145262	0.1416632	0.1420855	0.1407717	0.141673
2000 users								
50%	0.266	0.2642	0.226263	0.173654	0.171527	0.17034	0.169171	0.168771
70%	0.2304	0.2292	0.204045	0.163817	0.16154	0.161642	0.1597654	0.15936
90%	0.1699	0.1742	0.192715	0.1581	0.154478	0.15044	0.15273	0.151793
5000 users								
50%	0.2249	0.2241	0.250079	0.185797	0.183837	0.182474	0.179382	0.177451
70%	0.2288	0.2206	0.226694	0.1743	0.169922	0.169994	0.170496	0.169892
90%	0.2307	0.2324	0.210018	0.170393	0.167686	0.164501	0.164983	0.164337
10000 users								
50%	0.2615	0.2591	0.254009	0.189941	0.188926	0.185849	0.182247	0.181321
70%	0.2104	0.2073	0.23416	0.175754	0.174362	0.179101	0.172264	0.170842
90%	0.2354	0.234	0.210646	0.17097	0.167127	0.167633	0.165806	0.165159

Table 3: Prediction performance comparisons.

users given a web post, the other is to rank the web posts given a user.

4.6.1 User and Post Ranking Performance

We randomly select 30 web posts. For each post, at least 13 users should have shared it, i.e. at least 13 observed entries correspond to the post in \mathbf{X} . We hide 10 of the observed entries, and predict the entry values by HF-NMF, and then rank them in a descend order. The top-ranked users are regarded to be more influential with respect to the given post.

An example of user ranking is shown in Table 5. After converting the percentile social influence X_{ij} into \tilde{X}_{ij} by equation 3, the predicted social influence can be well interpreted. For example, given the post4405, we predict that if user1287 share the post, then 27 of her friends will visit it. In groundtruth, 26 of user1287’s friends visited the post after she shared the post. Also, given the post, the ranking of the users according to their social influence are mostly consistent with the groundtruth.

We conduct the similar process for post ranking. An example is shown in Table 4.

4.6.2 Comparison with Other Methods

In order to compare the proposed method with other comparative methods in ranking performance, we calculate the average η and ϱ over selected users and posts as the comparative measure. The results are shown in Table 6.

We can observe that HF-NMF achieves the best performance in both user ranking and post ranking tasks. bNMF+UF is better than bNMF+PF in user ranking, and bNMF+PF is better than bNMF+UF in post ranking, which demonstrate the effectiveness of the priors on users and posts. Also, by comparing the Table 3 and Table 6, we can see that the advantages of HF-NMF over other comparative methods are more obvious in ranking tasks.

5. CONCLUSION

In this paper, we study a novel problem of item-level social

	User	Ranking	Post	Ranking
	η	ϱ	η	ϱ
HF-NMF	0.8942	0.9389	0.8012	0.8697
bNMF+UF	0.8739	0.9088	0.7423	0.8334
bNMF+PF	0.8236	0.8412	0.7654	0.8548
bNMF	0.813	0.8342	0.7358	0.7926
AvgU	0.7824	0.8056	0.7047	0.7583
AvgP	0.6973	0.7143	0.6746	0.736
CoxPH	0.6596	0.6893	0.659	0.6762
LR	0.6524	0.697	0.6328	0.6593

Table 6: Comparison on user ranking.

influence prediction. We discover the effective user-specific and post-specific predictive factors, and propose a *Hybrid Factor Non-Negative Matrix Factorization* (HF-NMF) method to incorporate these predictive factors for user-post specific social influence prediction. To solve the problem, we devise an efficient Projected Gradient (PG) method for HF-NMF solution. Experimental results on a real social network dataset demonstrate that the proposed method can achieve better performance in social influence prediction compared with baseline methods. Meanwhile, we introduce the item-level social influence into users ranking and web posts ranking, and the experiment results show that the proposed method can effectively recommend the influential users given a web post, and the influential web posts given a user.

6. ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China, No. 60933013 and No. 61003097; National Basic Research Program of China, No. 2011CB302206; China Postdoctoral Science Foundation, No. 20100470285, and National Key Project Series, No. 2011ZX01042-001-002.

PostIDs	8783	9993	6551	8169	3550	8698	1404	5655	7825	4459
RankOrder(groundtruth)	1	2	3	4	5	6	7	8	9	10
SocialInfluence(groundtruth)	73	53	53	33	13	13	13	13	6	6
RankOrder(Prediction)	1	3	2	4	9	6	7	8	5	10
SocialInfluence(Prediction)	65	43	44	31	12	20	15	14	25	9

Table 4: A web posts ranking example for user 4852.

UserIDs	2627	1287	2336	2952	4466	2764	3052	0893	7666	4909
RankOrder(groundtruth)	1	2	3	4	5	6	7	8	9	10
SocialInfluence(groundtruth)	33	26	19	19	13	13	6	6	6	6
RankOrder(Prediction)	4	1	2	3	5	6	7	8	9	10
SocialInfluence(Prediction)	16	27	19	17	13	11	7	6	6	6

Table 5: A user ranking example for post 4405.

7. REFERENCES

- [1] N. Agarwal, H. Liu, L. Tang, and P. S. Yu. Identifying the influential bloggers in a community. In *Proceedings of the international conference on Web search and web data mining*, 2008.
- [2] A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.
- [3] E. Bakshy, B. Karrer, and L. A. Adamic. Social influence and the diffusion of user-created content. In *Proceedings of the 10th ACM conference on Electronic commerce*, 2009.
- [4] H. Bao and E. Y. Chang. Adheat: an influence-based diffusion model for propagating hints to match ads. In *Proceedings of the 19th international conference on World wide web*, 2010.
- [5] D. P. Bertsekas. *Nonlinear Programming*. MIT Press, MIT, 1999.
- [6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [7] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.
- [8] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. In *Proceedings of the third ACM international conference on Web search and data mining*, 2010.
- [9] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd International conference on Research and development in information retrieval*, 1999.
- [10] D. V. Kalashnikov, Z. Q. Chen, S. Mehrotra, and R. Nuray-Turan. Web people search via connection analysis. *IEEE Transactions on Knowledge and Data Engineering*, 20:1550–1565, 2008.
- [11] C. J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 19:2756–2779, 2007.
- [12] C. Macdonald and I. Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, 2006.
- [13] D. W. McMillan and D. M. Chavis. Sense of community: A definition and theory. *Journal of Community Psychology*, 14:6–23, 1986.
- [14] M. E. J. Newman. The structure and function of complex networks. *SIAM Reviews*, 45:167–256, 2003.
- [15] J. D. M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, 2005.
- [16] S. Rosset, C. Perlich, and B. Zadrozny. Ranking-based evaluation of regression models. *Knowledge and Information Systems*, 12:331–353, 2007.
- [17] S. H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, 2003.
- [18] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009.
- [19] M. M. Wasko and S. Faraj. Why should i share? examining social capital and knowledge contribution in electronic networks of practice. *Management Information System Quarterly*, 29:35–57, 2005.
- [20] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, 1994.
- [21] J. S. Weng, E. P. Lim, J. Jiang, and Q. He. Twiterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, 2010.
- [22] J. Yang and S. Counts. Predicting the speed, scale, and range of information diffusion in twitter. In *Proceeding of the international AAAI conference on Weblogs and social media*, 2010.