Expert Finding for Question Answering via Graph Regularized Matrix Completion

Zhou Zhao, Lijun Zhang, Xiaofei He and Wilfred Ng

Abstract—Expert finding for question answering is a challenging problem in Community-based Question Answering (CQA) systems, arising in many real applications such as question routing and identification of best answers. In order to provide high-quality experts, many existing approaches learn the user model from their past question-answering activities in CQA systems. However, the past activities of users in most CQA systems are rather few, and thus the user model may not be well inferred in practice. In this paper, we consider the problem of expert finding from the viewpoint of missing value estimation. We then employ users' social networks for inferring user model, and thus improve the performance of expert finding in CQA systems. In addition, we develop a novel graph-regularized matrix completion algorithm for inferring the user model. We further develop two efficient iterative procedures, GRMC-EGM and GRMC-AGM, to solve the optimization problem. GRMC-EGM utilizes the Extended Gradient Method (EGM), while GRMC-AGM applies the Accelerated proximal Gradient search Method (AGM), for the optimization. We evaluate our methods on the well-known question answering system Quora, and the popular social network Twitter. Our empirical study shows the effectiveness of the proposed algorithms in comparison to the state-of-the-art expert finding algorithms.

Index Terms—expert finding, community-based question answering, graph regularized matrix completion.

1 Introduction

The benefits of *Community-based Question Answering* (CQA) system have been well-recognized today. We have witnessed the popular CQA systems such as Yahoo Answer [1], Stack Overflow [2] and Quora [3]. Expert finding is an essential problem in CQA systems, which arises in many real applications such as question routing [42] and identification of best answers [5]. The central problem of expert finding is to choose the right users for answering the questions, which has attracted considerable attention recently in [5], [13], [41], [42], [48], [54], [53].

The process of the existing work [5], [54], [53], [13], [41], [42], [48] for the problem of expert finding in CQA systems can be divided into two steps: First, we build the user model from the past question-answering activities of the users, where the quality of the question-answering activities is voted by the CQA community via thumb-ups/downs. Second, we predict the performance of users for answering the new questions based on inferred user model, and then choose the users with highly predicted performance for answering the questions. Two most popular approaches for the problem of expert finding in CQA systems are authority-oriented algorithms and topic-oriented algorithms. The authority-oriented algorithms [5], [54], [53] construct the user-to-user graph based on ask-answer relations of the users, and then predict the performance for answering new questions based on user authority model. The topic-oriented algorithms [13], [41], [42], [48] are based on latent topic modeling techniques. They aim to fit a probabilistic model to the question-answering activities including both question content and the rating of users, and then use the latent user model to make further predictions.

Unlike previous studies, we consider the problem of expert finding from the viewpoint of missing value estimation. Given a rating matrix indicating the quality of users on answering existing questions, we want to choose the right experts for answering the new questions. Since the rating for users on answering some questions are unknown (i.e. missing values in the rating matrix), we want to predict the missing values in the rating matrix first, and then choose the users with highly predicted values for answering the new questions.

On the other hand, although existing expert finding methods have achieved promising performance, most of them have to obtain sufficient amount of past activities for inferring user model in order to provide high-quality experts. However, the past activities in most CQA system are rather few, and thus the user model may not be well inferred in practice. Fortunately, with the prevalence of online social networks today, it is not difficult to find the relation of CQA users, such as their connections in various online social networks (e.g., Facebook, Twitter, etc.). For example, we observe that more than one third of the users in Quora have a twitter account. A social relation between two users provides a strong evidence for them to have common preference or interest [22], [25], [55]. Thus, we strike for

- *Zhou Zhao, Wilfred Ng are with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, China. E-mail: zhaozhou@cse.ust.hk, wilfred@cse.ust.hk.*
- *Xiaofei He is with the State Key Lab of CAD&CG, College of Computer Science, Zhejiang University, 388 Yu Hang Tang Road, Hangzhou, Zhejiang 310058, China. E-mail: xiaofeihe@gmail.com.*
- *Lijun Zhang is with the Department of Computer Science and Technology, Nanjing University, Xianlin Campus Mailbox 603, 163 Xianlin Avenue, Qixia District, Nanjing 210023, China. E-mail: zljzju@gmail.com.*

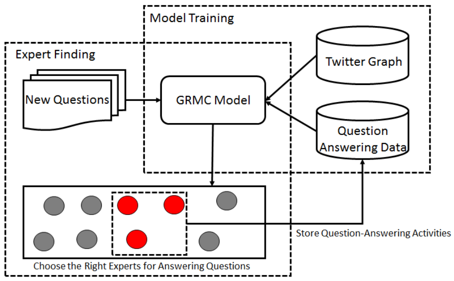


Fig. 1. Expert Finding in Community-based Question Answering System

integrating both user-to-user social relations and question-answering activities seamlessly into a common framework for tackling the problem of expert finding in CQA systems.

In this paper, we propose a new approach for the problem of expert finding in CQA system, which utilizes the online social relation of users, via graph regularized matrix completion. The rating matrix between questions and users keeps the quality of users on answering the questions. We consider that the content of the questions is the side information of the rating matrix. We choose the right experts for answering the questions by predicting the ratings of users on answering these questions. That is, we aim to complete the missing values for these questions in the rating matrix. We also employ the social relation of CQA users to regularize the completion of the rating matrix. If two users have strong connection in the social network, they may qualify for answering similar questions. We also develop two iterative procedures to solve the optimization problem. We illustrate the proposed expert finding algorithm in CQA System in Figure 1. When new questions come, our algorithm chooses the experts with highly predicted ratings for answering the questions and stores these question-answering activities into databases.

It is worthwhile to highlight several aspects of the proposed approach here:

1. We formulate the problem of expert finding in CQA systems from the viewpoint of missing value estimation. We predict the unknown values in the rating matrix via matrix completion, and choose the users with the highest predicted values for answering the new questions.

2. We integrate both the social relation of users and their past question-answering activities into one common framework for the problem of expert finding. We then propose the graph regularized matrix completion method for estimating the missing values in the rating matrix with social relation of users.

3. We represent the questions by *bag of words*, which has been shown to be successful in many question answering applications [12], [49], [52]. The user model is considered as a linear function which evaluates the qualification of the user for answering some question. That is, given a question, the expertise functions can predict the ratings for all users on answering this question.

The rest of the paper is organized as follows: Section 2 surveys the related work. We introduce the background and the problem of expert finding in Section 3. Next, we present the problem of expert finding from the viewpoint of graph regularized matrix completion in Section 4. We then provide a extended gradient method (GRMC-EGM) and an accelerated proximal gradient search method (GRMC-AGM) for solving the optimization problem in Section 5. A variety of experimental results are presented in Section 6. Finally, we provide some concluding remarks and suggestions for future work in Section 7.

2 Related Work

In this section, we briefly review some related work on expert finding and matrix completion.

2.1 Expert Finding

The existing work for the problem of expert finding can be categorized into two groups: the authority-oriented approach (cf. [5], [18], [23], [45], [53], [54]), and the topic-oriented approaches (cf. [11], [13], [14], [24], [28], [35], [41], [42], [48], [40], [27], [47], [10]).

The authority-oriented expert finding methods are based on link analysis for the ask-answer relation between users in the rating matrix. Thus, the user authority is ranked based on conventional web page ranking algorithms and their variations. For example, Bouguessa et al. [5] choose the experts to answer the questions based on the number of best answers provided by users, which is an In-degree-based method. Jurczyk et al. [18] construct the user-to-user graph from the past question-answering activities and employ a HITS [19] based method to rank the user authority. Zhu et al. [53], [54] measure the category relevance of questions and rank user authority in extended category link graph. Sung et al. [37] infer the expertise of new users by propagating the expertise of old users through common used words. Although authority-oriented expert finding methods can find the authoritative users, a new question might not match the the expertise of the global experts.

The topic-oriented expert finding methods are based on latent topic modeling techniques for the content of the questions. Deng et al. [11], Mimno et al. [28] and Hashemi et al. [14] develop latent user model for the problem of expert finding in DBLP bibliography. Guo et al. [13] and Zhou et al. [48] introduce the topic sensitive probabilistic approach to build the latent user model. Qu et al. [32] adopts PLSA model for analyzing the latent topic of questions to choose the right experts. Liu et al. [24] propose a probabilistic language model to predict the best answerer of the questions. Fatemeh et al. [35] build the model of users based on the topic modeling techniques. Xu et al. [41] propose a probabilistic dual role model by considering the asker role and answerer role of users. Weng et al. [40] choose the topic-sensitive influential users by leveraging topic models for answering the questions. McCallum et al. [27] devise an Author-Recipient-Topic model to take into account the topic distribution in the content posted by authors. Liu et al. [42] devise the CQArank model that estimates both the latent topic of questions and the user model. Zhao et al. [47] employ the topic models that generates the experts and the topic of the questions simultaneously. Chen et al. [10] model the expertise of the users based on the rating of the comments in community question answering system.

Unlike the previous studies, we formulate the problem of expert finding from the viewpoint of missing value estimation, which can be solved via matrix completion. We also employ the social relationship of users to improve the performance of missing value estimation.

2.2 Matrix Completion

The matrix completion techniques have been applied in many areas such as computer vision [30] and collaborative filtering [36], which is the process of adding missing entries to an incomplete matrix. Specifically, given the incomplete data matrix $\mathbf{Y} \in R^{m\times n}$ with low rank, the matrix completion problem is given by

$$\begin{aligned} \min_{\mathbf{X}} \quad & rank(\mathbf{X}) \\ s.t. \quad & Y_{ij} = X_{ij}, (i,j) \in \Omega, \end{aligned} \tag{1}$$

where $\mathbf{X} \in R^{m\times n}$ and Ω is the set of the observed entries. Unfortunately, the above rank minimization problem is NP-hard in general due to the nonconvexity and discontinuous nature of the rank function. Theoretical studies [34] show that the nuclear norm, i.e., the sum of singular values of a matrix, is the tightest convex lower bound of the rank function of matrices. Therefore, a widely used approach is to apply the nuclear norm as a convex surrogate of the nonconvex matrix rank function [26]:

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathbf{X}\|_* \\ s.t. \quad & Y_{ij} = X_{ij}, (i,j) \in \Omega, \end{aligned} \tag{2}$$

where $\|\mathbf{X}\|_* = \sum_{i=1}^{\min(m,n)} \delta_i(\mathbf{X})$ is the nuclear norm and $\delta_i(\mathbf{X})$ is the i-th largest singular value of $\mathbf{X}$. The existing approaches [8], [33], [7], [9], [6], [17], [43], [44] based on nuclear norm heuristic have provided theoretical guarantees and achieved excellent empirical performance. Currently, some efficient methods such as GreB [51], DCA [50], rank-one matrix pursuit algorithm [39] and conditional gradient method [15], [46] are proposed for matrix completion. However, our objective function is different from the objective function of matric completion and these methods may not be suitable for our problem.

3 The Problem of Expert Finding with Social Relationship

In this section, we first introduce some notations used in our subsequent discussion, which are the data matrix of questions $\mathbf{Q}$, the data matrix of users $\mathbf{X}$, the similarity matrix of users $\mathbf{W}$ and the observed rating matrix $\mathbf{Y}$. We then present the problem of expert finding in CQA systems from the viewpoint of missing value estimation.

TABLE 1
Summary of Notation

Notation	Notation Description
$\mathbf{Q}$	a data matrix of questions
$\mathbf{X}$	an expertise matrix of users
$\mathbf{Y}$	an observed rating matrix
$\mathbf{W}$	a similarity matrix of users
$\mathbf{L}$	a laplacian matrix of users
$\mathbf{D}$	a diagonal matrix of users
$\mathbf{\Omega}$	a set of existing ratings
$\mathbf{I}_\Omega$	an indicator matrix for observed ratings
$\mathbf{F}_1, \ldots, \mathbf{F}_n$	sets of following users
$f_{\mathbf{x}}(\mathbf{q}) = \mathbf{q}^{\mathbf{T}}\mathbf{x}$	an expertise function
λ_1	a social regularization term
λ_2	a nuclear norm regularization term

We represent the questions in CQA systems using bag-of-words model. Each question $\mathbf{q}_i$ is denoted by a d-dimensional word vector. We then denote the collection of questions by $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_m] \in R^{d\times m}$ where m is the total number of the questions.

We denote the collection of users in CQA systems by $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in R^{d\times n}$ where n is the total number of the users. The $\mathbf{x}_j$ is a d-dimensional vector for modeling the expertise of j-th users on word feature. The terms in $\mathbf{x}_j$ indicate the strengths and weakness of the j-th user on word feature of the questions. For example, consider that the i-th users is good at answering the questions containing the k-th word "algorithm", but is not good at answering the question with the l-th word "security". We then consider that the weight of the k-th word is higher than the l-th word in vector $\mathbf{x}_j$. That is, $x_{j,k} > x_{j,l}$.

We now propose a set of expertise functions for modeling the expertise of users in CQA systems. In this work, we use linear expertise functions for predicting the quality of users on answering the question, which is given by $f_{\mathbf{x}}(\mathbf{q}) = \mathbf{q}^{\mathbf{T}}\mathbf{x}$.

We observe that many of the CQA users also have connections on some online social networks. We then utilize the social relation of users to further improve our method. We can see that there exists two types of connections in different social networks, which are directed and undirected connections. The first type of connections is called social following (e.g., in Twitter), and the second type is called social friendship (e.g., in Facebook). We can integrate both types of the connections in our method.

We denote the similarity between users by $\mathbf{W} \in R^{n\times n}$. Now, we introduce the inference for the similarity matrix of users based on social friendship and social following, which is also used in [22], [25], [55]. First, consider that the connection between the i-th user and the j-th user is social friendship. The similarity $W_{ij} = 1$ when the i-th user and the j-th user are friends, otherwise, $W_{ij} = 0$. Then, consider that the connection between the i-th user and the j-th user is social following. Let $\mathbf{F}_i$ be the set of following users of the i-th user. We use the *Jaccard Distance* to model the similarity between the i-th user and

j-th user, which is $W_{ij} = \frac{|\mathbf{F}_i \bigcap \mathbf{F}_j|}{|\mathbf{F}_i \bigcup \mathbf{F}_j|}$. $\mathbf{F}_i \bigcap \mathbf{F}_j$ is the set of two users' common followings and $\mathbf{F}_i \bigcup \mathbf{F}_j$ is the set of two users' total followings. We note that the similarity value in $\mathbf{W}$ is within the range $[0,1]$.

We denote the quality of all users on answering the questions by the rating matrix $\mathbf{Y} \in R^{m\times n}$. The value in the rating matrix $\mathbf{Y}$ is voted by CQA community, which indicates the community's long term review for the quality of users on answering the questions. We notice that there are a number of missing values in $\mathbf{Y}$. Let $\mathbf{\Omega}$ be the set of existing ratings in $\mathbf{Y}$. The value Y_{ij} is observed if rating $(i,j) \in \mathbf{\Omega}$ exists in CQA systems.

We consider that the data matrix of questions $\mathbf{Q} \in R^{d\times m}$ is the side information of $\mathbf{Y} \in R^{m\times n}$. Therefore, the missing values in $\mathbf{Y}$ can be predicted by using the expertise function $f_{\mathbf{X}}(\mathbf{Q}) = \mathbf{Q}^{\mathbf{T}}\mathbf{X}$.

Using the notations above, we define the problem of expert finding in CQA systems as follows. Given the data matrix of questions $\mathbf{Q}$, the rating matrix of users $\mathbf{Y}$ and the similarity matrix of users $\mathbf{W}$, we aim to learn a expertise function $f_{\mathbf{x}}$ for each user $\mathbf{x}$ and complete the missing values in $\mathbf{Y}$. The best users qualifying for answering the question $\mathbf{q}$ are then selected according to $f_{\mathbf{x}}(\mathbf{q})$.

4 Expert Finding via Graph Regularized Matrix Completion

We first present the problem of expert finding from the viewpoint of missing value estimation, which can be solved by matrix completion. Then, we integrate the user-to-user social relationship to improve the quality of missing value estimation and propose a graph regularized matrix completion method for the problem.

4.1 Basic Objective Function

In this subsection, we propose a basic objective function for completing the missing values based on the past question-answering activities of users in CQA systems. Note that, the expertise of different users may be correlated. For example, two users with similar knowledge background are likely to be good at answering similar questions. Therefore, it is natural to assume that the expertise matrix $\mathbf{X}$ is of low rank. Consequently, we cast the problem of expert finding into the optimization problem of matrix completion [8], given by

$$\begin{aligned} &\min_{X} \|\mathbf{X}\|_* \\ s.t.\quad & Y_{ij} = f_{\mathbf{x}_j}(\mathbf{q}_i) = \mathbf{q}_i^T\mathbf{x}_j, \forall (i,j) \in \Omega \end{aligned} \tag{3}$$

where $\|\cdot\|_*$ stands for the trace (nuclear) norm of the data matrix for users $\mathbf{X}$ and $\mathbf{Y}$ is the rating matrix. By requiring $Y_{ij} = f_{\mathbf{x}_j}(\mathbf{q}_i)$, we expect that the learned expertise function $f_{\mathbf{x}_j}(\mathbf{q})$ can accurately estimate the qualification of user $\mathbf{x}_j$ for answering question $\mathbf{q}$.

Unlike the standard algorithm for matrix completion that requires solving an optimization problem involved the rating matrix $\mathbf{Y}$ of $n\times m$, the optimization problem given in Problem (3) only deals with the expertise matrix of users $\mathbf{X}$ with $n \times d$. Therefore it can be solved significantly more efficiently since $d \ll m$.

We also notice that the values in the rating matrix $\mathbf{Y}$ are noisy: first, humans may have bias on voting the quality of the answers; second, some answers are newly posted and it is difficult for them to receive a number of votes in a short time. However, the hard constraint in Problem (3) is not robust to the noise in $\mathbf{Y}$.

To overcome this limitation, we introduce the following optimization problem, given by

$$\min_{\mathbf{X}} \quad \|\mathbf{I}_\Omega \otimes (\mathbf{Y} - f_{\mathbf{X}}(\mathbf{Q}))\|_F^2 + \eta\|\mathbf{X}\|_* \tag{4}$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm, and $\otimes$ represents the Hadamard element-wise product. $\mathbf{I}_\Omega$ is an indicator matrix with ones for the observed ratings between questions and users, and zeros for the missing values. We keep $\|\mathbf{X}\|_*$ as the regularization terms in order to avoid the overfitting problem. The regularization coefficient η is employed to balance the weight between the data penalty term $\|\mathbf{I}_\Omega \otimes (\mathbf{Y} - f_{\mathbf{X}}(\mathbf{Q}))\|_F^2$ and the regularization term $\|\mathbf{X}\|_*$, which is usually set empirically.

4.2 Composite Objective Function

In this section, we present a composite objective function for estimating the missing value based on both the past question-answering activities of users in CQA systems and the social relation of users. The social relation between two users provides a strong evidence for them to have common bias [22], [25], [55]. Thus, we consider the social relation between users as a new regularization term for the optimization problem in Problem (4).

Consider the similarity matrix of users $\mathbf{W}$ which is inferred from social networks. Let $\mathbf{D}$ be the diagonal matrix with $D_{ii} = \sum_j W_{ij}$, and $\mathbf{L} = \mathbf{D} - \mathbf{W}$ be the Laplacian matrix. Based on the property of social relation, it is natural to require the similar users in the matrix $\mathbf{W}$ have similarity performance on question-answering activities. Thus, the new regularization on the data matrix of users $\mathbf{X}$ using the similarity matrix $\mathbf{W}$ can be achieved by minimizing [4]:

$$\begin{aligned} & \frac{1}{2}\sum_{k=1}^{m}\sum_{i,j=1}^{n} W_{ij}(f_{\mathbf{x}_i}(\mathbf{q}_k) - f_{\mathbf{x}_j}(\mathbf{q}_k))^2 \\ = & \frac{1}{2}\sum_k \left[\sum_{i,j} W_{ij}(\mathbf{q}_k^T\mathbf{x}_i - \mathbf{q}_k^T\mathbf{x}_j)^2\right] \\ = & \sum_k \left(\sum_i \mathbf{q}_k^T\mathbf{x}_i(\sum_j W_{ij})\mathbf{x}_i^T\mathbf{q}_k - \sum_{i,j}\mathbf{q}_k^T\mathbf{x}_i W_{ij}\mathbf{x}_j^T\mathbf{q}_k\right) \\ = & \sum_k \left(\sum_i \mathbf{q}_k^T\mathbf{x}_i D_{ii}\mathbf{x}_i^T\mathbf{q}_k - \sum_{i,j}\mathbf{q}_k^T\mathbf{x}_i W_{ij}\mathbf{x}_j^T\mathbf{q}_k\right) \\ = & \sum_k \mathbf{q}_k^T(\mathbf{X}\mathbf{D}\mathbf{X}^T - \mathbf{X}\mathbf{W}\mathbf{X}^T)\mathbf{q}_k \\ = & \sum_k \mathbf{q}_k^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{q}_k \\ = & tr\left(\mathbf{Q}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{Q}\right). \end{aligned} \tag{5}$$

We then obtain the following optimization problem on the expertise matrix of users $\mathbf{X}$

$$\begin{aligned} &\min_{\mathbf{X}} \|\mathbf{X}\|_* + \lambda tr(\mathbf{Q^T X L X^T Q}) \qquad (6) \\ &s.t. \quad Y_{ij} = f_{\mathbf{x}_j}(\mathbf{q}_i) = \mathbf{q}_i^T \mathbf{x}_j, \forall (i,j) \in \Omega \end{aligned}$$

where trace $tr(\cdot)$ represents the graph regularization for the expertise matrix of users $\mathbf{X}$, and $\lambda \geq 0$ is a tradeoff parameter.

Note that there is one fundamental difference between our formulation and the standard formulation of manifold regularization [4]. In our formulation $\mathbf{X}$ is the variable, whereas in the standard formulation $\mathbf{Q}$ is the variable.

Similarly, to tackle the problem of noisy values in the rating matrix $\mathbf{Y}$, we introduce the following optimization problem, given by

$$\begin{aligned} \min\nolimits_{\mathbf{X}} \quad & \|I_\Omega \otimes (\mathbf{Y} - f_{\mathbf{X}}(\mathbf{Q}))\|_F^2 + \lambda_1 tr\left(\mathbf{Q^T X L X^T Q}\right) \\ & + \lambda_2 \|\mathbf{X}\|_* \qquad (7) \end{aligned}$$

where $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$ are tradeoff parameters.

5 THE OPTIMIZATION

In this section, we design two simple but effective optimization methods to solve Problem (7).

We first have a brief discussion on the property of the composite objective function in Problem (7). We observe that this objective function falls into the general category composite optimization, which can be solved by Nesterov's gradient descent method [29]. Convex composite optimization refers to the convex optimization problem with the objective function formed as a sum of two terms: one is a smooth function, and another is a simple general convex function with known structure [29].

Let function $g(\mathbf{X}) = \|I_\Omega \otimes (\mathbf{Y} - f_{\mathbf{X}}(\mathbf{Q}))\|_F^2 + \lambda_1 tr\left(\mathbf{Q^T X L X^T Q}\right)$, which is the smooth part in our objective function, and $\lambda_2 \|\mathbf{X}\|_*$ is the non-smooth part. Therefore, the optimization Problem (7) is also given by

$$\min\nolimits_X \quad h(\mathbf{X}) = g(\mathbf{X}) + \lambda_2 \|\mathbf{X}\|_*. \qquad (8)$$

Since the trace norm term $\|\mathbf{X}\|_*$ in Problem (8) is non-smooth, a natural approach for solving this problem is Nesterov's gradient descent method which needs to evaluate the gradient of the smooth part. Thus, in order to solve the convex composition optimization Problem (8), we need to evaluate the gradient of $g(\mathbf{X})$ below.

Let $\mathbf{e}_j \in \{0,1\}^n$ be the j-th united vector. The gradient of $g(\mathbf{X})$ is given by

$$\nabla g(\mathbf{X}) = \sum_{(i,j) \in \Omega} 2(\mathbf{q}_i^T \mathbf{x}_j - Y_{ij}) \mathbf{q}_i \mathbf{e}_j^T + 2\lambda_1 \mathbf{Q Q^T X L}, \quad (9)$$

where can be verified in [31].

We then introduce a very useful tool, that is, the singular value shrinkage operator [6].

Definition 1: (Singular Value Shrinkage Operator) Consider the singular value decomposition (SVD) of a matrix $\mathbf{X} \in R^{m \times n}$ of rank r,

$$\mathbf{X} = \mathbf{U \Sigma V}^T, \mathbf{\Sigma} = diag(\{\delta_i\}_{1 \leq i \leq r}). \qquad (10)$$

Define the singular value shrinkage operator D_τ as follows:

$$D_\lambda(\mathbf{X}) = \mathbf{U} D_\lambda(\mathbf{\Sigma}) \mathbf{V}^T \qquad (11)$$

and

$$D_\lambda(\mathbf{\Sigma}) = diag(\{\max\{0, \delta_i - \lambda\}\}). \qquad (12)$$

Using the singular value shrinkage operator above, we have the following useful theorem for composite objective function below:

Theorem 1: ([6]) For each $\lambda \geq 0$ and $\mathbf{C} \in R^{m \times n}$, we have

$$D_\lambda(\mathbf{C}) = \arg\min_X \frac{1}{2} \|\mathbf{X} - \mathbf{C}\|_F^2 + \lambda \|\mathbf{X}\|_*. \qquad (13)$$

We can see that the singular value shrinkage operator of the matrix $\mathbf{C}$ is the solution to the composite optimization Problem (13).

Therefore, we solve Problem (8) based on the property of composite optimization function and singular value shrinkage operator. We choose the variant of Nesterov's gradient descent method in [17], which is explicitly designed for trace norm minimization. We present the extend gradient method (EGM) and accelerated gradient method (AG-M) [17] for our objective function with graph regularized nuclear norm in the following sections, respectively.

5.1 The Optimization Using GRMC-EGM

In this section, we present an extended gradient method to solve the graph regularized matrix completion Problem (8), denoted by GRMC-EGM.

The optimization Problem (8) ($\min_X h(\mathbf{X})$) can also be reformulated under the framework of proximal regularization [38] and can be solved iteratively, given by

$$\mathbf{X} \quad = \quad \arg\min_X G_{\rho_k}(\mathbf{X}, \mathbf{X}_{k-1}), \qquad (14)$$

where

$$\begin{aligned} G_{\rho_k}(\mathbf{X}, \mathbf{X}_{k-1}) \quad &= \quad g(\mathbf{X}_{k-1}) + \frac{\rho_k}{2} \|\mathbf{X} - \mathbf{X}_{k-1}\|_F^2 \\ &+ \quad tr\left((\mathbf{X} - \mathbf{X}_{k-1})^T \nabla g(\mathbf{X}_{k-1})\right) \\ &+ \quad \lambda_2 \|\mathbf{X}\|_* \qquad (15) \end{aligned}$$

and ρ_k is a positive scalar.

Given the initial setting $\mathbf{X}_0$, ρ_0 and γ, Problem (14) can be solved via the following two steps alternatively.

Computing $\mathbf{X}_k$: Fix ρ_{k-1}, and minimize $G_{\rho_{k-1}}(\mathbf{X}, \mathbf{X}_{k-1})$ for $\mathbf{X}_k$

Ignoring constant terms, $G_{\rho_{k-1}}(\mathbf{X}, \mathbf{X}_{k-1})$ can be rewritten as

$$\begin{aligned} & G_{\rho_{k-1}}(\mathbf{X}, \mathbf{X}_{k-1}) \\ = \quad & \frac{\rho_{k-1}}{2} \left\| \mathbf{X} - \left(\mathbf{X}_{k-1} - \frac{1}{\rho_{k-1}} \nabla g(\mathbf{X}_{k-1}) \right) \right\|_F^2 + \lambda_2 \|\mathbf{X}\|_* \\ & \qquad (16) \end{aligned}$$

We notice that Equation 16 can fall into the general composite objective function in Theorem 1. We consider

Algorithm 1 The optimization using GRMC-EGM

Input: $\mathbf{Y}$, $\mathbf{L}$, $\mathbf{Q}$ and tolerance ε
Initialize: ρ_0, γ and $\mathbf{X}_0$

repeat
  STEP 1. Update $\mathbf{X}_k$ as
  $\mathbf{X}_k = D_{\frac{\lambda_2}{\rho_{k-1}}}\left(\mathbf{X}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{X}_{k-1})\right)$
  STEP 2. Update ρ_k as
  $\hat{\rho} \leftarrow \rho_{k-1}$
  while $h(\mathbf{X}_k) > G_{\hat{\rho}}(\mathbf{X}_k, \mathbf{X}_{k-1})$ **do**
    $\hat{\rho} = \gamma\hat{\rho}$
  $\rho_k \leftarrow \hat{\rho}$
until $\|\mathbf{X}_k - \mathbf{X}_{k-1}\|_F < \varepsilon$

that matrix $\mathbf{C} = \mathbf{X}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{X}_{k-1})$. Thus, we can obtain the closed form solution of the above Problem (14) by Theorem 1 as follows:

$$\mathbf{X}_k = D_{\frac{\lambda_2}{\rho_{k-1}}}\left(\mathbf{X}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{X}_{k-1})\right) \tag{17}$$

Computing ρ_k: We then compute the step size $\frac{1}{\rho_k}$ properly such that convergence rate of our method can be $O(\frac{1}{k})$. We first introduce the following useful theorem to learn the convergence rate:

Theorem 2: ([6]) Let $\mathbf{X}^* = \arg\min_X h(\mathbf{X})$ and ρ be the smoothing parameter. Then for any $k \geq 1$, we have

$$h(\mathbf{X}_k) - h(\mathbf{X}^*) \leq \frac{\gamma\rho\|\mathbf{X}_0 - \mathbf{W}^*\|_F^2}{2k} \tag{18}$$

if we can find an appropriate value for ρ at each iteration such that the condition

$$h(\mathbf{X}_k) \leq G_\rho(\mathbf{X}_k, \mathbf{X}_{k-1}) \tag{19}$$

is satisfied.

Fix $\mathbf{X}_{k-1}$, we compute the value of ρ_k with multiplier γ to satisfy the condition in Theorem 2 at each iteration, given by

$$\hat{\rho} \leftarrow \rho_{k-1}, \text{ and } \hat{\rho} = \gamma\hat{\rho} \tag{20}$$

and $G_\rho(\mathbf{X}_k, \mathbf{X}_{k-1})$ is then update by Equation 15.

The whole procedure of GRMC-EGM is summarized in Algorithm 1. The main computation cost of GRMC-EGM in each iteration is the computation of SVD in **STEP 1**. The additional cost in **STEP 2** is much smaller. For large scale problems, we can adopt some existing techniques [21] to accelerate the computation of SVD and make Algorithm 1 more efficient. Furthermore, the convergence of Algorithm 1 for Problem (14) is $O(\frac{1}{k})$, which is guaranteed by the EGM [17].

5.2 The Optimization Using GRMC-AGM

In this section, we introduce an accelerated gradient method for solving the regularized graph matrix completion Problem (8), given by GRMC-AGM.

Algorithm 2 The optimization using GRMC-AGM

Input: $\mathbf{Z}$, $\mathbf{L}$, $\mathbf{Q}$ and tolerance ε
Initialize: ρ_0, γ and $\mathbf{X}_0$

repeat
  STEP 1. Update $\mathbf{X}_k$ as
  $\mathbf{X}_k = D_{\frac{\lambda_2}{\rho_{k-1}}}\left(\mathbf{Z}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{Z}_{k-1})\right)$
  STEP 2. Update α_k as
  $\alpha_k = \frac{1+\sqrt{1+4\alpha_{k-1}^2}}{2}$
  STEP 3. Update $\mathbf{Z}_k$ as
  $\mathbf{Z}_k = \mathbf{X}_{k-1} + \frac{\alpha_{k-1}-1}{\alpha_k}(\mathbf{X}_k - \mathbf{X}_{k-1})$
  STEP 4. Update ρ_k as
  $\hat{\rho} \leftarrow \rho_{k-1}$
  while $h(\mathbf{Z}_k) > G_{\hat{\rho}}(\mathbf{Z}_k, \mathbf{Z}_{k-1})$ **do**
    $\hat{\rho} = \gamma\hat{\rho}$
  $\rho_k \leftarrow \hat{\rho}$
until $\|\mathbf{X}_k - \mathbf{X}_{k-1}\|_F < \varepsilon$

Firstly, we introduce a new variable $\mathbf{Z}$. GRMC-AGM method constructs an approximation of $h(\mathbf{X})$ at a given point $\mathbf{Z}$ as

$$\begin{aligned} G_\rho(\mathbf{X}, \mathbf{Z}) &= g(\mathbf{Z}) + \frac{\rho_k}{2}\|\mathbf{X} - \mathbf{Z}\|_F^2 \\ &+ tr\left((\mathbf{X} - \mathbf{Z})^T\nabla g(\mathbf{Z})\right) \\ &+ \lambda_2\|\mathbf{X}\|_* \end{aligned} \tag{21}$$

Then, GRMC-AGM method solves the optimization Problem (8) by iteratively updating $\mathbf{X}$, $\mathbf{Z}$, and ρ.

Computing $\mathbf{X}_k$: In the kth iteration, we update $\mathbf{X}_k$ as the unique minimizer of $G_{\rho_{k-1}}(\mathbf{X}, \mathbf{Z}_{k-1})$:

$$\begin{aligned} \mathbf{X}_k &= \arg\min_X G_{\rho_{k-1}}(\mathbf{X}, \mathbf{Z}_{k-1}) \\ &= \frac{\rho_{k-1}}{2}\left\|\mathbf{X} - \left(\mathbf{Z}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{Z}_{k-1})\right)\right\|_F^2 \\ &+ \lambda_2\|\mathbf{X}\|_* \end{aligned} \tag{22}$$

Similarly, we consider that the matrix $\mathbf{C} = \mathbf{Z}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{X}_{k-1})$. Then, we can obtain the closed form solution of the above Problem (22) by Theorem 1 as follows:

$$\mathbf{X}_k = D_{\frac{\lambda_2}{\rho_{k-1}}}\left(\mathbf{Z}_{k-1} - \frac{1}{\rho_{k-1}}\nabla g(\mathbf{Z}_{k-1})\right) \tag{23}$$

Computing $\mathbf{Z}_k$: Then, $\mathbf{Z}_k$ is updated in the same way as [17]:

$$\alpha_k = \frac{1+\sqrt{1+4\alpha_{k-1}^2}}{2}, \tag{24}$$

$$\mathbf{Z}_k = \mathbf{X}_{k-1} + \frac{\alpha_{k-1}-1}{\alpha_k}(\mathbf{X}_k - \mathbf{X}_{k-1}) \tag{25}$$

We summarize the main procedure for solving problem 8 by AGM in Algorithm 2. Compared with GRMC-EGM, Algorithm 2 has a convergence rate of $O(\frac{1}{k^2})$, which is guaranteed by the convergence property of the AGM method [17].

TABLE 2
SUMMARY OF DATASETS

Data	Quora						Twitter	
	Question Type	# Questions	# Users	# Answers	Rating Ratio	Average Ratings	# Edges	#Edge Ratio
D_1	What	176,114	37,408	387,822	5.89×10^{-5}	5.5743	36,602	2.62×10^{-5}
D_2	Who	12,387	13,325	27,368	0.0002	5.2192	9,242	5.21×10^{-5}
D_3	Why	38,414	23,369	73,687	8.21×10^{-5}	6.7333	13,188	2.41×10^{-5}
D_4	When	7,101	7,252	12,011	0.0002	3.562	4,562	8.67×10^{-5}
D_5	Where	12,261	12,628	21,779	0.0001	2.382	7,926	4.97×10^{-5}
D_6	Which	15,521	17,913	35,519	0.0001	3.991	11,688	3.64×10^{-5}
D_7	How	69,563	35,520	122,659	4.96×10^{-5}	4.5794	21,542	1.71×10^{-5}
D_8	General Question	101,685	43,943	189,165	4.23×10^{-5}	4.0799	25,416	1.32×10^{-5}

6 EXPERIMENTAL RESULTS

In this section, we conduct several experiments on the question-answering platform, Quora, and the social network, Twitter to show the effectiveness of our proposed approaches (GRMC-GEM and GRMC-AGM) for the problem of expert finding in CQA systems. The experiments are conducted by using Matlab, tested on machines with Linux OS Intel(R) Core(TM2) Quad CPU 2.66Hz, and 32GB RAM.

6.1 Data Preparation

We collect the data from a popular question answering system, Quora, where questions are posted and answered by its community users. Quora was launched to the public in June, 2010 and has become very successful in just a few years. We crawled the questions posted between September 2012 and August 2013. We also crawled all the users who answered these questions. In total, we collect 444,138 questions, 95,915 users and 887,771 answers.

We first classify the collected questions into eight categories based on *head word* feature such as “what”, “who” and “is/are”. The *head word* feature is widely used for question classification in [16], [20]. We access the performance of our methods on different categories of questions. We remove the questions which have no or only one answer.

We next collect the rating for users on answering the questions through thumbs-up/down. The summary of the collected thumbs-up/down value with their counts is shown in Figure 2. We observe that the thumbs-up/down count distribution is a power-law distribution with means most thumbs-up/down are relatively small, which are between 0 and 100. We thus normalize the value of thumbs-up/down to the range of $[0, 100]$.

We then sort the collected questions based on their posted timestamp. We use the second quartile of the questions (first 50%) as training data and consider the third quartile of the questions (second 50%) for testing. So training and testing data do not have overlap. This validation process is also used in [42]. We observe that more than one third of the users in Quora have Twitter account. We first extract the Twitter account of the Quora users, then crawl their following relationship from Twitter graph. We then build the similarity matrix of users based on their following relation in Twitter. The summary of the collected dataset can be found in Table 2.

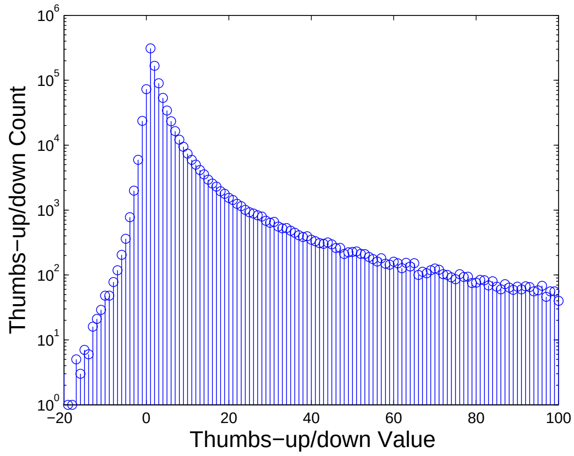


Fig. 2. Summary of Thumbs-up/down Value

6.2 Evaluation Criteria

We evaluate the performance of our proposed methods based on four popular evaluation criteria for the problem of expert finding in CQA systems, i.e. Mean Reciprocal Rank (MRR) [53], [54], [23], normalized Discounted Cumulative Gain (nDCG) [42], [23], Precision@1 [53], [54], [13] and Accuracy [48], [41].

For ground truth, we consider all the answerers for each question as the target user set, and their received thumbs-up/down as the ground truth rating scores. The experts for the questions tend to get more thumbs-up [42]. Note that our task is not to predict the exact thumbs-up/down value of each user but rank them in terms of thumbs-up/down value.

Given the testing question set $\mathcal{Q}$, we introduce the four evaluation criteria below.

MRR. We denote by r_{best}^q the rank of the best answerer for question q by an algorithm. MRR measures the ranking quality for the best answerer by an algorithm. The MRR measure is given by

$$MRR = \frac{1}{|\mathcal{Q}|}\sum_{q\in\mathcal{Q}}\frac{1}{r_{best}^q}.$$

TABLE 3
EXPERIMENTAL RESULTS ON MRR (THE **best** SCORE IN **bold**)

MRR	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
AuthorityRank	0.5524	0.5582	0.6032	0.5935	0.5734	0.5569	0.5852	0.5863
ExpertsRank	0.5445	0.5575	0.588	0.5772	0.5552	0.5383	0.5761	0.5774
TSPM	0.5715	0.5629	0.6023	0.6155	0.5953	0.5688	0.6099	0.5918
DRM	0.6055	0.6017	0.6315	0.6363	0.6246	0.596	0.6487	0.6404
CRAR	0.6107	0.6257	0.6596	0.664	0.6227	0.6057	0.6457	0.646
GRMC-EGN	0.6554	0.6455	0.6767	0.6722	0.6663	0.6315	0.6150	**0.6797**
GRMC-AGM	**0.6687**	**0.6659**	**0.6786**	**0.7067**	**0.6722**	**0.6317**	**0.6770**	0.6723

TABLE 4
EXPERIMENTAL RESULTS ON NDCG (THE **best** SCORE IN **bold**)

nDCG	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
AuthorityRank	0.8627	0.8529	0.8634	0.8853	0.8804	0.8601	0.8695	0.8726
ExpertsRank	0.8592	0.8487	0.8556	0.8782	0.8769	0.8575	0.8645	0.8649
TSPM	0.8528	0.8318	0.8523	0.8857	0.883	0.823	0.8598	0.8734
DRM	0.8616	0.8543	0.8689	0.8902	0.8888	0.8513	0.8848	0.8798
CRAR	0.8811	0.8524	0.8756	0.9088	0.8942	0.8736	0.8861	0.8876
GRMC-EGN	0.9117	0.8913	0.8890	0.9127	0.9292	0.8848	0.9138	0.9283
GRMC-AGM	**0.9289**	**0.9084**	**0.8936**	**0.9369**	**0.9474**	**0.9153**	**0.9263**	**0.9658**

where $|\mathcal{Q}|$ is the number of testing questions used.

nDCG. We denote by R^q the ranking order of all the users for question q by an algorithm. R_i^q is user who is ranked on the i-th position by an algorithm. $|R^q|$ is the number of ranked users for question q. We consider that the relevance between question q and user j is indicated by thumbs-up/down value Y_{qj}.

The Discounted Cumulative Gain (DCG) for the ranked users of question q is given by

$$DCG = Y_{qR_1^q} + \sum_{i=2}^{|R_i^q|} \frac{Y_{qR_i^q}}{\log_2 i},$$

and the normalized Discounted Cumulative Gain is given by

$$nDCG = \frac{DCG}{IDCG},$$

where IDCG is the DCG of ideal ordering. We report the average nDCG for all datasets.

Precision@1. We use Precision@1 to measure the ranking quality of the best answerer, given by

$$Precision@1 = \frac{|\{q \in \mathcal{Q} | r_{best}^q = 1\}|}{|\mathcal{Q}|}.$$

In other words, Precision@1 computes the average number of times that the best answerer is ranked on top by an algorithm.

Accu. We employ Accu to access the ranking quality of the best answerer. The Accu measure is normalized by the number of answerers for a question, which is given by

$$Accu = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \frac{|R^q| - r_{best}^q}{|R^q| - 1},$$

where Accu = 1 (best) means that the best answerer returned by an algorithm always ranks on top while Accu = 0 means the opposite.

In summary, **MRR**, **Precision@1** and **Accu** are different measures for the ranking quality of the best answerer by an algorithm. **nDCG** is the measure for the ranking quality of all answerers by an algorithm.

6.3 Performance Evaluations and Comparisons

We compare our proposed method with other five popular expert finding algorithms in CQA systems as follows:

- **ExpertsRank** algorithm [45]: This ExpertsRank algorithm uses question ask-answer relation to construct the graph of users, and then finds the experts with link structure analysis based on PageRank algorithm.
- **AuthorityRank** algorithm [5]: The AuthorityRank algorithm computes user authority based on the number of provided best answers, which is an in-degree method.
- **TSPM** algorithm [13]: The TSPM algorithm is a topic-sensitive probabilistic method for expert finding in CQA systems, which fits a LDA-based probabilistic model to the question-answering activities.
- **DRM** algorithm [41]: The DRM algorithm is also a topic-sensitive probabilistic method for expert finding in CQA, systems which fits a PLSA-based probabilistic model to the question-answering activities.
- **CRAR** algorithm [54]: The CRAR algorithm ranks user authority based on link analysis based on both target question category and its relevant categories for expert finding using topic model.

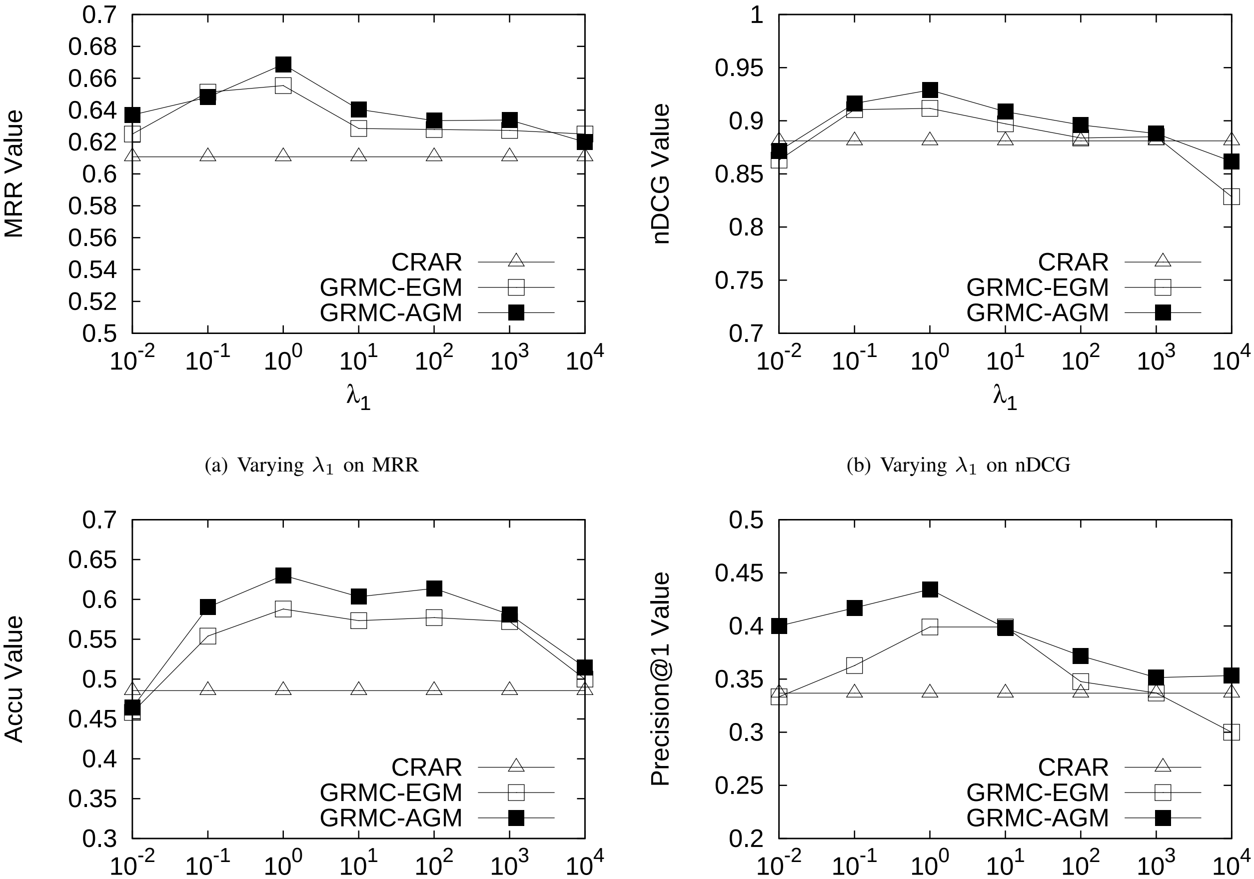
Table 3, 4, 6 and 5 show the evaluation results on MRR, nDCG, Precision@1 and Accu, respectively. The evaluation were conducted with different categories of the questions from D_1 to D_8. For each dataset, we report the performance of all methods in the tables.

TABLE 5
EXPERIMENTAL RESULTS ON ACCU (THE **best** SCORE IN **bold**)

ACCU	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
AuthorityRank	0.4397	0.4512	0.4584	0.4179	0.4113	0.442	0.4296	0.4448
ExpertsRank	0.4316	0.4437	0.4488	0.3993	0.3982	0.423	0.423	0.4374
TSPM	0.4445	0.4489	0.4485	0.4287	0.4195	0.4378	0.4383	0.4365
DRM	0.4416	0.4434	0.4408	0.4097	0.4003	0.433	0.4292	0.4393
CRAR	0.4855	0.5029	0.4996	0.4696	0.4411	0.4733	0.4729	0.4869
GRMC-EGN	0.5733	0.533	0.5607	0.5634	0.5165	0.5327	0.5618	**0.5983**
GRMC-AGM	**0.6301**	**0.5357**	**0.6624**	**0.6064**	**0.5836**	**0.5785**	**0.5988**	0.5892

TABLE 6
EXPERIMENTAL RESULTS ON AVERAGE PRECISION@1 (THE **best** SCORE IN **bold**)

Precision@1	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
AuthorityRank	0.3112	0.3282	0.3702	0.3521	0.3263	0.3158	0.341	0.3376
ExpertsRank	0.3009	0.3276	0.352	0.3367	0.3042	0.2992	0.3324	0.3336
TSPM	0.3082	0.3448	0.3757	0.3525	0.3239	0.3212	0.3468	0.3375
DRM	0.3217	0.3141	0.3433	0.3442	0.3235	0.3152	0.3612	0.3535
CRAR	0.3368	0.3673	0.3896	0.4263	0.3441	0.3356	0.3663	0.3715
GRMC-EGN	0.399	0.3688	0.394	0.4175	0.392	0.3818	0.4166	0.4266
GRMC-AGM	**0.4345**	**0.3715**	**0.4**	**0.4615**	**0.4177**	**0.3964**	**0.4328**	**0.442**



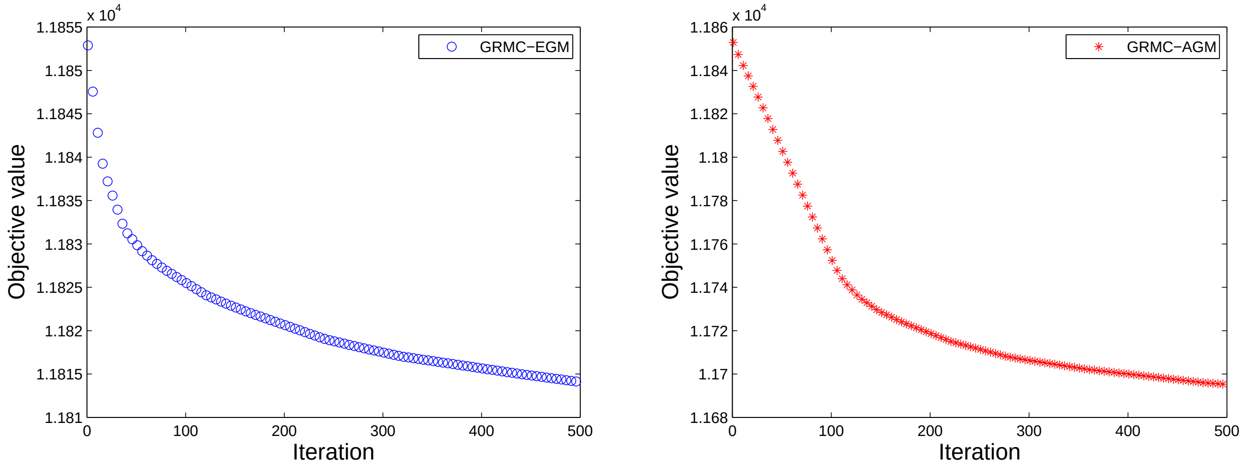
(a) Varying λ_1 on MRR

(b) Varying λ_1 on nDCG

(c) Varying λ_1 on Accu

(d) Varying λ_1 on Precision@1

Fig. 3. The performance of GRMC-EGM and GRMC-AGM versus parameter λ_1.



(a) GRMC-EGM (b) GRMC-AGM

Fig. 4. The convergence of GRMC-EGM and GRMC-AGM

The AuthorityRank and ExpertsRank methods are based on link analysis of users while TSPM and DRM methods are based on probabilistic model. The CRAR method is based on both link analysis of users and probabilistic model. These experiments reveal a number of interesting points:

- The topic-oriented methods, both TSPM and DRM, outperform the AuthorityRank and ExpertsRank methods, which suggests the effectiveness of the latent user model for the problem of expert finding.
- The CRAR method achieves better performance of other baseline methods. This suggests that the link analysis of users can also improve the performance of expert finding.
- In all the cases, our GRMC-EGM and GRMC-AGM methods achieve the best performance. This shows that leveraging the power of both the social network of users and the viewpoint of missing value estimation can further improve the performance of expert finding.

In our approaches, there is an essential parameter, that is, the graph regularization parameter λ_1. When the value of λ_1 becomes small, our problem can be considered as the original problem of matrix completion, which is only based on the past question-answering activities. We vary parameter λ_1 to investigate the benefits of our methods from the idea of graph regularized matrix completion for missing value estimation based on social network of users. We vary the value of parameter λ_1 from 10^{-4} to 10^4 and show the evaluation results in Figures 3(a), 3(b), 3(c) and 3(d).

We notice that the CRAR method consistently outperforms other methods in most of the datasets. Thus, we mainly compare our methods with CRAR method on data D_1 by varying parameter λ_1. The performance trend of our methods by varying parameter λ_1 is similar on other data sets.

As we can see, the performance of both GRMC-EGM nd GRMC-AGM methods is very stable with respect to the parameter λ_1. Both GRMC-EGM and GRMC-AGM methods almost achieve consistently good performance when λ_1 varies from 10^{-2} to 10^4 in Figures 3(a), 3(b), 3(c) and 3(d).

As we have described, both GRMC-EGM and GRMC-AGM use the social network of users to capture the local geometric structure of the user model. The success of graph regularized matrix completion for expert finding relies on the assumption that two neighboring users share the similar user model. When the parameter λ_1 is small, the formulation of graph regularized matrix completion can be considered as matrix completion for expert finding without graph of users. We observe that the performance of both GRMC-EGM and GRMC-AGM is relatively low, which is similar to the performance of DRM (based on PLSA model). This is the reason why the performance of both GRMC-EGM and GRMC-AGM methods increases as λ_1 increases, as shown in Figures 3(a), 3(b), 3(c) and 3(d).

6.4 Convergence Study

The updating rules for minimizing the objective function of both GRMC-EGM and GRMC-AGM methods are essentially iterative. Here we investigate how both GRMC-EGM and GRMC-AGM methods converge.

Figures 4(a) and 4(b) show the convergence curves of GRMC-EGM and GRMC-AGM methods on data D_1, respectively. The y-axis is the value of the objective function and x-axis denotes the iteration number. We can observe that method GRMC-AGM converges much faster than method GRMC-EGM.

7 Conclusions

We formulated the problem of expert finding from a new perspective of missing value estimation. We presented a novel method called graph regularized matrix completion for estimating missing values in the rating matrix for the problem of expert finding. We consider that the user model is the expertise function which can be learned from the rating matrix and past question-answering activities. Furthermore, our approach integrates both the social relation of users and their past question-answering activities seamlessly into a common framework for the problem of expert finding in CQA systems. In this way, our approach can further improve the estimation for the missing values in the rating matrix for finding the experts. We devise two simple but efficient iterative methods, i.e., GRMC-EGM and GRMC-AGM to solve the optimization problem for graph regularized matrix completion, respectively. We conduct several experiments on the data collected from the famous question-answering system, Quora. The experimental results demonstrate the advantage of the GRMC-based algorithms with five state-of-the-art expert finding methods. In the future, we will explore the kernel expertise function as the user model for the problem of expert finding.

Acknowledgement We would like to express our thanks to the editor and the reviewers for their careful revisions and insightful suggestions.

References

[1] Yahoo Answer: http://answers.yahoo.com/.

[2] Stack Overflow: http://stackoverflow.com/.

[3] Quora: http://www.quora.com/.

[4] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[5] M. Bouguessa, B. Dumoulin, and S. Wang. Identifying authoritative actors in question-answering forums: the case of yahoo! answers. In *SIGKDD*, pages 866–874. ACM, 2008.

[6] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[7] E. J. Candes and Y. Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[8] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[9] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *Information Theory, IEEE Transactions on*, 56(5):2053–2080, 2010.

[10] B.-C. Chen, J. Guo, B. Tseng, and J. Yang. User reputation in a comment rating environment. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 159–167. ACM, 2011.

[11] H. Deng, I. King, and M. R. Lyu. Formal models for expert finding on dblp bibliography data. In *ICDM*, pages 163–172. IEEE, 2008.

[12] A. Figueroa and G. Neumann. Learning to rank effective paraphrases from query logs for community question answering. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[13] J. Guo, S. Xu, S. Bao, and Y. Yu. Tapping on the potential of q&a community by recommending answer providers. In *CIKM*, pages 921–930. ACM, 2008.

[14] S. H. Hashemi, M. Neshati, and H. Beigy. Expertise retrieval in bibliographic network: a topic dominance learning approach. In *CIKM*, pages 1117–1126. ACM, 2013.

[15] E. Hazan. Sparse approximate solutions to semidefinite programs. In *LATIN 2008: Theoretical Informatics*, pages 306–316. Springer, 2008.

[16] Z. Huang, M. Thint, and Z. Qin. Question classification using head words and their hypernyms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 927–936. Association for Computational Linguistics, 2008.

[17] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 457–464. ACM, 2009.

[18] P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *CIKM*, pages 919–922. ACM, 2007.

[19] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[20] F. Li, X. Zhang, J. Yuan, and X. Zhu. Classifying what-type questions by head noun tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 481–488. Association for Computational Linguistics, 2008.

[21] M. Li and J. T.-Y. Kwok. Making large-scale nystrom approximation possible. 2010.

[22] W.-J. Li and D.-Y. Yeung. Relation regularized matrix factorization. In *IJCAI*, pages 1126–1131, 2009.

[23] J. Liu, Y.-I. Song, and C.-Y. Lin. Competition-based user expertise score estimation. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 425–434. ACM, 2011.

[24] X. Liu, W. B. Croft, and M. Koll. Finding experts in community-based question-answering services. In *CIKM*, pages 315–316. ACM, 2005.

[25] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296. ACM, 2011.

[26] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128(1-2):321–353, 2011.

[27] A. McCallum, A. Corrada-Emmanuel, and X. Wang. Topic and role discovery in social networks. *Computer Science Department Faculty Publication Series*, page 3, 2005.

[28] D. Mimno and A. McCallum. Expertise modeling for matching papers with reviewers. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 500–509. ACM, 2007.

[29] Y. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

[30] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2233–2246, 2012.

[31] K. B. Petersen and M. S. Pedersen. The matrix cookbook. *Technical University of Denmark*, pages 7–15, 2008.

[32] M. Qu, G. Qiu, X. He, C. Zhang, H. Wu, J. Bu, and C. Chen. Probabilistic question recommendation for question answering communities. In *Proceedings of the 18th international conference on World wide web*, pages 1229–1230. ACM, 2009.

[33] B. Recht. A simpler approach to matrix completion. *The Journal of Machine Learning Research*, 7777777:3413–3430, 2011.

[34] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[35] F. Riahi, Z. Zolaktaf, M. Shafiei, and E. Milios. Finding expert users in community question answering. In *Proceedings of the 21st international conference companion on World Wide Web*, pages 791–798. ACM, 2012.

[36] R. Salakhutdinov and N. Srebro. Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. *tc (X)*, 10:2, 2010.

[37] J. Sung, J.-G. Lee, and U. Lee. Booming up the long tails: Discovering potentially contributive users in community-based question answering services. In *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.

[38] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization*, 6(615-640):15, 2010.

[39] Z. Wang, M.-j. Lai, Z. Lu, W. Fan, H. Davulcu, and J. Ye. Rank-one matrix pursuit for matrix completion. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 91–99, 2014.

[40] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. ACM, 2010.

[41] F. Xu, Z. Ji, and B. Wang. Dual role model for question recommendation in community question answering. In *Proceedings of SIGIR*, pages 771–780. ACM, 2012.

[42] L. Yang, M. Qiu, S. Gottipati, F. Zhu, J. Jiang, H. Sun, and Z. Chen. Cqarank: jointly model topics and expertise in community question answering. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 99–108. ACM, 2013.

[43] J. Yi, R. Jin, A. K. Jain, and S. Jain. Crowdclustering with sparse pairwise labels: A matrix completion approach. In *AAAI Workshop on Human Computation*, volume 2, 2012.

[44] J. Yi, R. Jin, S. Jain, and A. Jain. Inferring users preferences from crowdsourced pairwise comparisons: A matrix completion approach. In *First AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[45] J. Zhang, M. S. Ackerman, and L. Adamic. Expertise networks in online communities: structure and algorithms. In *WWW*, pages 221–230. ACM, 2007.

[46] X. Zhang, D. Schuurmans, and Y.-l. Yu. Accelerated training for matrix-norm regularization: A boosting approach. In *Advances in Neural Information Processing Systems*, pages 2906–2914, 2012.

[47] T. Zhao, N. Bian, C. Li, and M. Li. Topic-level expert modeling in community question answering. In *SDM*, pages 776–784. SIAM, 2013.

[48] G. Zhou, S. Lai, K. Liu, and J. Zhao. Topic-sensitive probabilistic model for expert finding in question answer communities. In *Proceedings of CIKM*, pages 1662–1666. ACM, 2012.

[49] G. Zhou, Y. Liu, F. Liu, D. Zeng, and J. Zhao. Improving question retrieval in community question answering using world knowledge. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2239–2245. AAAI Press, 2013.

[50] T. Zhou, W. Bian, and D. Tao. Divide-and-conquer anchoring for near-separable nonnegative matrix factorization and completion in high dimensions. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 917–926. IEEE, 2013.

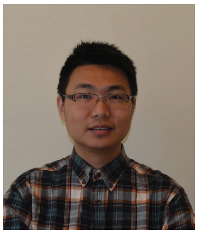
[51] T. Zhou and D. Tao. Greedy bilateral sketch, completion & smoothing. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pages 650–658, 2013.

[52] T. C. Zhou, X. Si, E. Y. Chang, I. King, and M. R. Lyu. A data-driven approach to question subjectivity identification in community question answering. In *AAAI*, 2012.

[53] H. Zhu, H. Cao, H. Xiong, E. Chen, and J. Tian. Towards expert finding by leveraging relevant categories in authority ranking. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2221–2224. ACM, 2011.

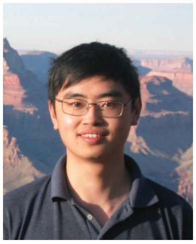
[54] H. Zhu, E. Chen, H. Xiong, H. Cao, and J. Tian. Ranking user authority with relevant knowledge categories for expert finding. *World Wide Web*, pages 1–27, 2013.

[55] J. Zhu, H. Ma, C. Chen, and J. Bu. Social recommendation using low-rank semidefinite program. In *AAAI*, 2011.



Zhou Zhao received his BS degree in computer science from the Hong Kong University of Science and Technology (HKUST), in 2010. He is currently a PhD student in the Department of Computer Science and Engineering, HKUST. His research interests include machine learning and data mining.

Lijun Zhang Lijun Zhang received the B.S. and Ph.D. degrees in Software Engineering and Computer Science from Zhejiang University, China, in 2007 and 2012, respectively. He is currently an associate professor of the Department of Computer Science and Technology, Nanjing University, China. Prior to joining Nanjing University, he was a postdoctoral researcher at the Department of Computer Science and Engineering, Michigan State University, USA. His research interests include machine learning, optimization, information retrieval and data mining.



Xiaofei He Xiaofei He received the B.S. degree in computer science from Zhejiang University, Hangzhou, China, in 2000, and the Ph.D. degree in computer science from the University of Chicago, Chicago, IL, USA, in 2005. He is currently a Professor with the College of Computer Science, Zhejiang University. Prior to joining Zhejiang University, he was a Research Scientist with Yahoo! Research Labs, Burbank, CA, USA.

Wilfred Ng received his MSc (Distinction) and PhD in Computer Science from the University of London. Currently he is an Associate Professor of Computer Science and Engineering at the Hong Kong University of Science and Technology, where he is a member of the database research group. His research interests are in the areas of databases, data mining and information Systems, which include Web data management and XML searching. Further Information can be found at the following URL: http://www.cs.ust.hk/faculty/wilfred/index.html.