Collaborative Filtering with Social Local Models

Huan Zhao, Quanming Yao[1], James T. Kwok and Dik Lun Lee
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Hong Kong
{hzhaoaf, qyaoaa, jamesk, dlee}@cse.ust.hk

Abstract—**Matrix Factorization (MF) is a very popular method for recommendation systems. It assumes that the underneath rating matrix is low-rank. However, this assumption can be too restrictive to capture complex relationships and interactions among users and items. Recently, Local LOw-rank Matrix Approximation (LLORMA) has been shown to be very successful in addressing this issue. It just assumes the rating matrix is composed of a number of low-rank submatrices constructed from subsets of similar users and items. Although LLORMA outperforms MF, how to construct such submatrices remains a big problem. Motivated by the availability of rich social connections in today's recommendation systems, we propose a novel framework, i.e., Social LOcal low-rank Matrix Approximation (SLOMA), to address this problem. To the best of our knowledge, SLOMA is the first work to incorporate social connections into the local low-rank framework. Furthermore, we enhance SLOMA by applying social regularization to submatrices factorization, denoted as SLOMA++. Therefore, the proposed model can benefit from both social recommendation and the local low-rank assumption. Experimental results from two real-world datasets, Yelp and Douban, demonstrate the superiority of the proposed models over LLORMA and MF.** [1]

Index Terms—**Recommendation system, Collaborative Filtering, Matrix factorization, Local low-rank, Social network**

I. INTRODUCTION

Recommendation System (RS) has become an indispensable tool in the big data era. It tackles information overload by helping users to get interesting items based on their previous behaviors. Collaborative Filtering (CF), a state-of-the-art RS technique, tries to predict users' ratings (or preferences) on unseen items based on similar users or items. Among various CF-based methods, Matrix Factorization (MF) is most popular due to its good performance and scalability [1], [2], [3], [4]. MF is based on the assumption that users' preferences to items are controlled by a small number of latent factors. Thus, the large user-item rating matrix can be decomposed into two smaller matrices, representing user-specific and item-specific latent factors, respectively. In other words, the rating matrix is of *low-rank*.

Despite the success of MF in RS, the assumption that the rating matrix is low-rank (termed *global low-rank*) is problematic because the rating matrix in a real-world scenario is very large and composed of diverse rating behaviors, making the low-rank assumption of the rating matrix improper. Lee et al. [5] proposed a novel framework called Local LOw-Rank

Matrix Approximation (LLORMA) to alleviate the global low-rank problem. Based on the observation that there tend to be groups of users who are interested in small sets of items, LLORMA takes the view that the rating matrix is composed of a number of low-rank submatrices (termed *local low-rank*), which is illustrated in the right part of Figure 1. Extensive experiments have demonstrated the effectiveness of LLORMA in recommendation systems [5], [6], [7], [8]. Besides RS, subsequent works have been performed in different domains based on the local low-rank assumption, e.g., image processing [9], [10], multi-label classifications [11], document analysis [12], demonstrating the efficacy of this framework.

In the local low-rank approach, the construction of the submatrices is a fundamental problem. Lee et al. [5] proposed to firstly choose some random anchor points, i.e., user-item pairs, from the rating matrix. Then, for each anchor point, one submatrix is constructed by selecting the remaining points that are close to the anchor point based on some distance metrics. However, this method leads to several problems. Firstly, since anchor points are randomly selected, it is hard to explain the meaning of the obtained submatrices, and thus the recommendation results are also unexplainable. Secondly, since the submatrices are constructed around the anchor points using a distance threshold, they are very sensitive to the distance threshold. Thus, finding a good distance threshold value is a challenging task. Finally, the computation and space costs are both high, because we need to compute and store the pair-wise similarities of users and items in all submatrices.

To overcome the weaknesses of LLORMA, we propose a novel framework, Social LOcal low-rank Matrix Approximation (SLOMA), which incorporates the social connections among users into the local low-rank framework. While social connections have been exploited effectively in MF-based RSs [13], [14], [15], [16], [17], to the best of our knowledge, SLOMA is the first work to utilize social connections under the local low-rank framework. As in most social network research, SLOMA assumes that the social graph embeds a number of social groups, within which users have similar preferences to and influences on each other. The basic idea of SLOMA is illustrated in the left part of Figure 1. There are three social groups underlying the social network,[2] based on which we can construct three submatrices out of the rating matrix

---

[1]Quanming Yao is the Corresponding author.

[2]Note that the groups can overlap in reality, here we put them separately for simplicity.
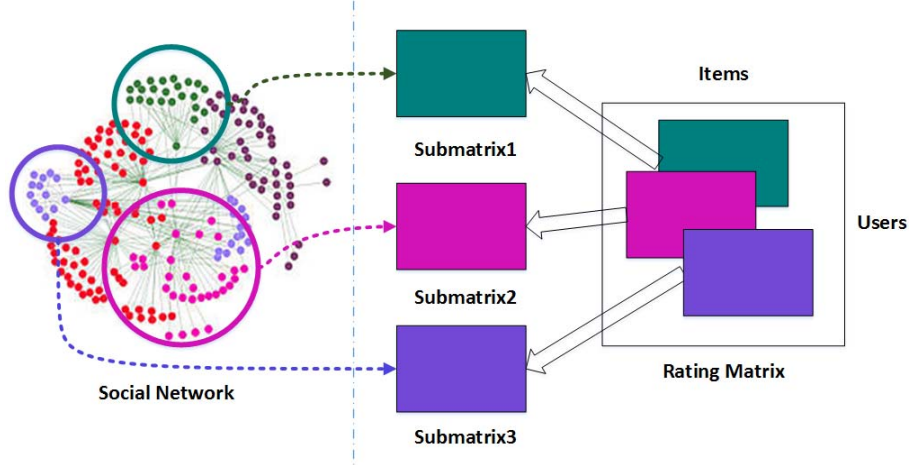
Fig. 1. An illustration of SLOMA. **Left-hand side:** an example social network, where underlying social groups are circled. **Right-hand side:** the core idea of LLORMA, that is, the rating matrix is composed of a number of submatrices which are low-rank. SLOMA enhances LLORMA with the social groups underlying the social network to construct meaningful submatrices. Therefore, SLOMA can enjoy the advantages of both social recommendation and LLORMA, leading to better recommendation performance.

satisfying the low-rank assumption. The social connections help SLOMA in solving the important submatrix construction problem. Specifically, instead of randomly selecting anchor points as in LLORMA, SLOMA can select influential users, termed *connectors*, and pick the connectors' friends within a certain distance in the social graph (i.e., within several hops) to construct the submatrices, which are meaningful because they can be treated as social groups. See the example in Figure 1. Thus, SLOMA solves all the three problems faced by LLORMA, that is, it selects meaningful anchors and build meaningful submatrices, its distance measure is intuitive, and there is no need to keep the pairwise similarities of users and items for each submatrix, which avoids high computation and space costs.

In summary, the contributions of our work are as follows.

- To the best of our knowledge, SLOMA is the first work that incorporates social connections into the local low-rank framework, which can enjoy the advantages from both sides, i.e., social recommendation and local low-rank framework.
- SLOMA addresses the submatrice construction problem in LLORMA by exploiting social connections. These submatrices have better interpretability (explained by social homophily theory) and lead to superior prediction accuracy over LLORMA.
- In addition to SLOMA, we also propose SLOMA++ that incorporates social regularization techniques into each local model to further improve the performance.
- We conduct extensive experiments on two real-world datasets to show the effectiveness of our proposed methods. We not only demonstrate the effectiveness of SLOMA and SLOMA++, but also give insights into social recommendation under the local low-rank framework.

The rest of the paper is organized as follows. MF, LLORMA

and social recommendation are introduced in Section II. We then elaborate our SLOMA model in Section III, and the experiments as well as the analysis in Section IV and V, respectively. Finally, we conclude our work in Section VI.

II. BACKGROUND

A. MF and LLORMA

MF has been one of the most popular approaches for rating prediction in recommendation systems. Based on the assumption that users' preferences to items are governed by a small number of latent factors, the rating matrix $\mathbf{O} \in \mathbb{R}^{m \times n}$ can be approximated by the product of l-rank matrices,

$$\mathbf{R} \approx \mathbf{U}\mathbf{V}^{\top}, \qquad (1)$$

where $\mathbf{U} \in \mathbb{R}^{m \times l}$ and $\mathbf{V} \in \mathbb{R}^{n \times l}$ with $l \ll min(m,n)$, representing, respectively, the latent features of users' preferences and items. The approximating process can be completed by solving the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2} \sum_{(i,j) \in \Omega} (\mathbf{O}_{ij} - \mathbf{u}_i \mathbf{v}_j^{\top})^2, \qquad (2)$$

where $\mathbf{u}_i$ and $\mathbf{v}_j$ are the ith and jth rows of $\mathbf{U}$ and $\mathbf{V}$, representing the latent feature vectors of user u_i and item v_j, respectively. $|| \cdot ||_F$ denotes the Frobenius norm, and Ω is the index set of the observations, and $\mathbf{O}_{ij}$ is the observed rating of user u_i to item v_j. In order to avoid overfitting, two regularization terms, $\frac{\lambda}{2} \left(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2 \right)$, are added to Equation (2). In the literature, this method is also termed Regularized SVD (RegSVD) [1].

Instead of assuming the rating matrix to be low-rank, Lee et al. proposed a novel framework LLORMA that assumes the rating matrix is *local low-rank* [5], i.e., it is composed of a number of submatrices which are of low-rank. After obtaining the submatrices, MF is applied to each submatrix

independently, and then a weighted ensemble scheme is designed to approximate the rating matrix. The key component of LLORMA is the construction of submatrices, which is depicted in Algorithm 1.

Algorithm 1 Submatrice Construction (LLORMA)

Input: Observed index set Ω, number of submatrices q, distance threshold d_1, d_2;

1: **for** $t = 1, 2, ..., q$ **do**
2:     Select a point (i_t, j_t) randomly from Ω;
3:     Obtain a subset of users $U^t = \left\{ u_k : d(u_{i_t}, u_k) \leq d_1 \right\}$;
4:     Obtain a subset of items $V^t = \left\{ v_k : d(v_{j_t}, v_k) \leq d_2 \right\}$;
5:     Obtain $\mathbf{M}^t$; // *The tth submatrix based on U^t and V^t.*
6: **end for**
7: **return** $\left\{ \mathbf{M}^t \right\}_{t=1}^q$

In Line 2, a point (i_t, j_t), termed *anchor point*, is chosen randomly.[3] $d(\cdot)$ is a distance function for any two users or items in the rating matrix. d_1 and d_2 are two distance thresholds for selecting users or items according to Line 3 and 4. A submatrix is then constructed from the selected users and items that are within the distance threshold from the anchor point. In LLORMA, the distance d is based on the cosine similarity between any two latent features of users or items. It is reported that such measurement gives the best predicting performance [5]. Specifically, the arc-cosine between users u_{i_t} and u_k is

$$d(i_t, k) = \arccos \left(\frac{\mathbf{u}_{i_t} \mathbf{u}_k}{||\mathbf{u}_{i_t}|| \cdot ||\mathbf{u}_k||} \right). \tag{3}$$

Besides, the similarity between items are also computed based on arc-cosine distance at Equation (3).

After obtaining q submatrices, MF is applied to each submatrice independently, leading to q groups of user-specific and item-specific latent features. Finally, the rating matrix $\mathbf{R}$ is approximated by the weighted ensemble of product of these q groups of user-specific and item-specific latent feature vectors as follows

$$\mathbf{R}_{ij} = \sum_{t=1}^q \frac{w_{ij}^t}{\sum_{s=1}^q w_{ij}^s} \left[\mathbf{u}_i^t (\mathbf{v}_j^t)^\top \right], \tag{4}$$

where w_{ij}^t is the weight computed based on the distance between a point (u_i, v_j) and an anchor point (u_{i_t}, v_{j_t}) in every submatrix. $\mathbf{u}_i^t$ and $\mathbf{v}_j^t$ represent the latent feature vectors of u_i and v_j, respectively, from the tth submatrix. It means that the prediction of a point (u_i, v_j) in the rating matrix is a weighted average of the predicted ratings from all submatrices where u_i and v_j both occur. Experiments on real-world datasets show that LLORMA improves prediction accuracy compared to RegSVD. Besides recommendation [5], [6], [7], [8], the local low-rank assumption has been demonstrated to be effective in many other domains, including image processing [9], [10], multi-label classifications [11], and document analysis [12].

---

[3]There are more complex methods for picking up anchor points, but random choosing gives the best predicting performance as reported in [5]. Thus, we focus on random choosing here.

B. Social Recommendation in MF

Social recommendation has become more and more popular with the proliferation of online social networks such as Facebook, Twitter and Yelp. Previous works have been performed to incorporate social connections into the MF framework [13], [14], [15], [17], [18]. Most of these works are based on the social homophily theory [19] that people with similar preferences tend to be connected as friends. Thus, the latent feature vectors of friends should be closer after factorizing the rating matrix. One typical method is to add the social regularization term to the objective function (2), which is defined in the following:

$$\frac{\beta}{2} \sum_{i=1}^m \sum_{j \in \mathcal{F}(i)} \mathbf{S}_{ij} ||\mathbf{u}_i - \mathbf{u}_j||_2^2, \tag{5}$$

where $\beta > 0$ is the weight of social regularization term, $\mathcal{F}(i)$ is the set of u_i's friends, and $\mathbf{S}_{ij}$ is the similarity between u_i and u_f. The more similar two users are, the closer their latent feature vectors.

We note that there are no previous work on social recommendation under the local low-rank framework. The major difference between our SLOMA and LLORMA is that LLORMA uses random anchor points to construct submatrices, while we use social connections. Thus, we can obtain meaningful submatrices and better prediction ability because each submatrix can represent the behavior of a socially connected user group. Moreover, we enhance SLOMA to SLOMA++ by incorporating social regularization into every local model, and demonstrate with experiments that recommendation performance can be further improved.

III. SOCIAL LOCAL MATRIX APPROXIMATION

In this section, we first describe the main problem facing submatrix construction in LLORMA, which also motivates our work. Then, we elaborate the proposed SLOMA framework.

A. Motivation

Despite the success of LLORMA, the main weakness of LLORMA is its submatrix construction method (Algorithm 1). We identify three problems below.

First, the anchor points are chosen randomly, which do not have any reasonable justification and cannot be interpreted. Second, the constructed submatrices are neither stable nor meaningful. Specifically, the cosine similarity in Equation (3) is used in LLROMA, which is computed in Lines 3 and 4 of Algorithm 1. From Lemma III.1 (proof can be found in Appendix A), we can see that different pairs of $(\mathbf{U}, \mathbf{V})$ may be obtained from solving the optimization problem in Equation (2). However, the cosine similarity is not robust to the transformation in Lemma III.1. Using the same anchor point with the pair $(\mathbf{U}, \mathbf{V})$ or $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$, different submatrices can be constructed. As a result, the constructed submatrices are not consistent and then become meaningless. Third, to predict a missing rating in the original big rating matrix, a weighted average scheme is used to combine the ratings in the

submatrices according to Equation (4). This suffers from the same problem facing submatrix construction, as the weights are also computed based on cosine similarity.

Lemma III.1. *Given any matrices $\mathbf{U}$ and $\mathbf{V}$ which are an optimal solution to (2), then $\hat{\mathbf{U}} = \mathbf{U}\mathbf{Q}$ and $\hat{\mathbf{V}} = \mathbf{V}\mathbf{Q}^{-1}$ are also an optimal solution, if matrix $\mathbf{Q}$ is invertible.*

Motivated by these problems, we propose a novel framework integrating social connections with the local low-rank framework. Based on the social homophily theory [19] that people with similar preferences tend to be connected as friends, we use the social connections among users as an explicit indicator of similarity between users' preferences in constructing the submatrices. Since each submatrix contains socially connected users with similar preferences, it satisfies the low-rank property. Moreover, we argue and demonstrate with experiments that the submatrices obtained in our model are better than those in LLORMA in terms of overall prediction ability. The core idea of our proposed model is illustrated in Figure 1.

B. SLOMA

In this section, we give a formal and mathematical elaboration of SLOMA and design the optimization approaches.

We first introduce the notations in this work. Let $\mathcal{U} = \{u_1, u_2, ..., u_m\}$ and $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ be the sets of users and items, respectively. Let $\mathcal{G} = (\mathcal{U}, \mathcal{E})$ represents a social network graph, where the vertex set $\mathcal{U} = \{u_1, u_2, ..., u_m\}$ represents the users and the edge set $\mathcal{E} = \{e_1, e_2, ..., e_p\}$ represents the social connections among all users. The weight of every edge of $\mathcal{G}$ is set to 1, representing the existence of friendship between two users.[4] Then, for any two users u_i, u_j, $d(u_i, u_j)$ is the distance between u_i and u_j, which can be computed from the social graph. In this paper, we propose a novel framework to integrate social recommendation with the local low-rank assumption. The key challenge is how to construct submatrices from the rating matrix. We develop several approaches, including heuristic and systematic approaches.

After constructing the submatrices, we apply MF to them and obtain a number of user-specific and item-specific latent features in different submatrices. It is obvious that these submatrices have overlaps, i.e., a user and an item can occur in more than one submatrices, which aligns with our intuition that users tend to participate in multiple groups in social networks.

Overall, SLOMA consists of the following steps:

- Identify q social groups from the social graph $\mathcal{G}$, and then construct q submatrices from the rating matrix $\mathbf{O}$ based on those groups.
- Apply MF to all the submatrices, i.e., $\{\mathbf{M}^t\}_{t=1}^q$, independently, and obtain q groups of user-specific and item-specific latent features.
- Predict the missing ratings in $\mathbf{R}$ using the ensemble of predictions from all submatrices.

[4]We assume that the social graph is undirected and unweighted.

C. Construction of Submatrices

In SLOMA, we assume that there are groups of users underlying the social network, whose preferences are similar due to the fact that they are socially connected. To identify these groups, we develop both heuristic and systematic approaches in this work.

Heuristic Approaches. For the heuristic approaches, we build the social groups based on the fact that users' influences to each other can propagate through the networks. We observe that users in a social group can affect each other and the amount of influence can vary. Intuitively, those with more friends tend to have larger influence than others. Based on this, we construct the submatrices by first selecting a number of influential users, i.e., the connectors, and their friends within a fixed number of hops, e.g., three, in the social network. Then, for each group of users, we select the items they rate and then the submatrix is constructed from this user-item group. Note that there are several methods for selecting the connectors with different overall performance. We discuss this in Section III-E and analyze the experimental results in Section V-D. After selecting q connectors, construction of the submatrix is shown in Algorithm 2. The shortest path in Line 3 can be obtained by algorithms such as Dijkstra's algorithm.

Algorithm 2 Heuristic Submatrix Construction.

Input: Observed index set Ω, social graph $\mathcal{G} = (\mathcal{U}, \mathcal{E})$, number of submatrices q, distance threshold d;
1: Obtain q users as connectors from $\mathcal{U}$;
2: **for** $t = 1, 2, \ldots, q$ **do**
3:     Obtain $D(u_{i_t}, u_k)$: the shortest distance between u_{i_t} and all the other users u_ks in $\mathcal{G}$;
4:     Obtain a subset of users $U^t = \{u_k : D(u_{i_t}, u_k) \leq d\}$;
5:     Obtain a subset of items $V^t = \{v_j : (i, j) \in \Omega, \forall u_i \in U^t\}$; // *All items rated by users in U^t.*
6:     Obtain $\mathbf{M}^t$; // *The tth submatrix based on U^t and V^t*
7: **end for**
8: **return** $\{\mathbf{M}^t\}_{t=1}^q$

Systematic Approaches. The systematic approaches are based on methods for overlapping community detection, which have been intensively investigated in recent years. We can see that social groups are equivalent to communities in social networks. In the literature, a community is a group of people who have more interactions within the group than those outside it [20], and hence users in the same community have more characteristics in common than with users outside of it. This leads to the low-rank property of the submatrix constructed from one community. Naturally, communities can overlap with each other, and many works have been done for detecting such communities, e.g., line graph partitioning [21], clique percolation [22], eigenvector methods [23], egonet analysis [24], [25] and low-rank models [26]. In this work, we adopt *BIGCLAM* [26], which can scale to large datasets and has good empirical performance. The process is shown in Algorithm 3. We can see

that the only difference from Algorithm 2 is how we construct the social groups (Line 1).

Algorithm 3 Systematic Submatrix Construction

Input: Observed index set Ω, social graph $\mathcal{G} = (\mathcal{U}, \mathcal{E})$, number of submatrices q;

1: Apply BIGCLAM [26] to $\mathcal{G}$ to obtain the set C of q communities;
2: **for** $t = 1, 2, \ldots, q$ **do**
3:   Obtain a subset of users $U^t \in C$; // U^t *is the tth community.*
4:   Obtain a subset of items $V^t = \{v_j : (i, j) \in \Omega, \forall u_i \in U^t\}$; // *All items rated by users in U^t.*
5:   Obtain $\mathbf{M}^t$; // *The tth submatrix based on U^t and V^t*
6: **end for**
7: **return** $\{\mathbf{M}^t\}_{t=1}^q$

Note that different methods lead to different numbers of users in the submatrices, i.e., they may cover different numbers of users in the social graph, which will influence the overall prediction ability of SLOMA. In Sections V-D and V-E, we give detailed comparisons and analysis for these different submatrix construction methods, and show that the heuristic approaches outperform the systematic ones.

D. Matrix Factorization in Social Local Models

After obtaining q submatrices, we apply MF to each submatrix independently according to Equation (2). Note that in LLORMA, each submatrix corresponds to a local model. Similarly, each submatrix in SLOMA corresponds to a social local model. In the following sections, we use the terms "local model" and "submatrix" interchangeably. By training each social local model, we can obtain q groups of user latent feature matrices, $\mathbf{U}^1, \ldots, \mathbf{U}^q$, and item latent feature matrices, $\mathbf{V}^1, \ldots, \mathbf{V}^q$. Then, the overall rating of user u_i given to item v_j is predicted according to the following ensemble method:

$$\mathbf{R}_{ij} = \frac{1}{q} \sum_{t=1}^q \mathbf{u}_i^t (\mathbf{v}_j^t)^\top, \tag{6}$$

where $\mathbf{u}_i^t$ and $\mathbf{v}_j^t$, respectively, are the user-specific and item-specific latent feature vectors in the tth submatrix where they occur, and q is the number of submatrices where user u_i and item v_j both occur. Equation (6) means that the rating of u_i given to v_j is predicted by the average of predicted ratings that u_i gives to v_j in all submatrices.

E. Different Methods to Select Connectors

The construction of social groups in SLOMA is similar to overlapping community detection using seed set expansion [27], [28]. There tend to be a small number of so-called long-tail users, i.e., those with few friends, which are difficult to be selected into any of the social groups, leading to incomplete coverage of SLOMA. We define two kinds of coverage in this work: i) User Coverage is the percentage of unique users in $\mathcal{U}$ selected by all of the social groups;

ii) Rating Coverage is the percentage of unique ratings in Ω selected by all of the submatrices.

Rating coverage is dependent on user coverage because when constructing each submatrix, we usually select all of the items rated by that group of users. Therefore, in this paper, we use coverage to represent any of these two concepts depending on the context. In the local low-rank framework, the coverage of the submatrices is very important for the overall prediction performance. In LLORMA, the random selection of anchor points and choice of distance threshold are used to control the rating coverage. Due to the randomness of the anchor points, the coverage of all submatrices will be 1 when the number of anchor points reaches a large enough value. Further, the larger the threshold is, the larger the coverage of each submatrix.

In SLOMA, we select seeds, i.e., connectors, in the social graph and propagate from each seed along different number of hops to construct the social groups. For long-tail users who cannot be covered, we predict their ratings with the mean of all the observed ratings, which is the same approach as in [5]. In Section V-A, we can see that the performance of SLOMA is still better than that of LLORMA and RegSVD, demonstrating the superiority of the submatrix construction method in SLOMA. However, it is still very important to cover as many points in the rating matrix as possible. Thus, we try several connector selection methods, which are described below:

- **Hub**: Select a set of users with the largest number of neighbors.
- **Random**: Randomly select a set of users.
- **Random-Hub**: It is an integration of the above two methods, that is, we first select a larger number of hub users, e.g., 1000, and then randomly select a smaller number of the hub users, e.g., 50.
- **Greedy**: Each time we select a connector, we select from those not yet covered by connectors that are already selected.

After obtaining the set of connectors, we construct one submatrix around each connector user as shown in Lines 4 and 5 of Algorithm 2. In Section V-D, we compare the performance of these methods as well as the community-based submatrix construction method. We can see that despite the existence of a small number of uncovered users, Hub and Greedy obtain the best performance comparing to the other methods.

F. SLOMA++

In addition to constructing the submatices in SLOMA, social connections can be used to enhance each local model by applying social regularization to the factorization of each submatrix. Social regularization adds constraints that users with direct social connections should be closer in the latent space, which is introduced in Equation (5).

For each local model, the social regularization term is defined as:

$$\frac{\beta}{2} \sum_{i=1}^m \sum_{j \in \mathcal{F}(i)} \mathbf{S}_{ij}^t ||\mathbf{u}_i^t - \mathbf{u}_j^t||_2^2, \tag{7}$$

where $\mathbf{S}^t \in \mathbb{R}^{|U^t| \times |U^t|}$ is the similarity matrix with $\mathbf{S}^t_{ij}$ representing the similarity between users u_i and u_j in the tth submatrix, and $|U^t|$ is the number of users in the tth submatrix. Note that we apply Equation (7) when each submatrix is factorized. We term this model SLOMA++.

To calculate the similarities between two users u_i and u_j, we utilize the popular Person Correlation Coefficient (*PCC*) [29]:

$$\mathbf{S}^t_{ij} = \frac{\sum\limits_{f \in T(i,j)} (\mathbf{M}^t_{if} - \bar{M}^t_i)(\mathbf{M}^t_{jf} - \bar{M}^t_j)}{\sqrt{\sum\limits_{f \in T(i,j)} (\mathbf{M}^t_{if} - \bar{M}^t_i)^2} \sqrt{\sum\limits_{f \in T(i,j)} (\mathbf{M}^t_{jf} - \bar{M}^t_j)^2}},$$

where $\bar{M}^t_i$ represents the mean value of all of the ratings of user u_i in the tth submatrix $\mathbf{M}^t$, and $I^t(i)$ represents the set of items rated by user u_i in the tth submatrix $\mathbf{M}^t$. $T(i,j) = I^t(i) \cap I^t(j)$, representing the set of common items rated by u_i and u_j in the tth submatrix. Note that when constructing the submatrix for a group of users, we take all of the group's ratings in $\mathbf{O}$. Thus, for the same user pair u_i and u_j, $T(i,j)$ will be the same across all submatrices. This is why we remove the superscript t from $T(i,j)$. Further, we employ a mapping function $g(x) = \frac{1}{2}(x+1)$ to bound the range of *PCC* similarity to $[0,1]$.

For each submatrix, we need to solve the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2} \sum_{(i,j) \in \Omega} (\mathbf{O}_{ij} - \mathbf{u}_i \mathbf{v}_j^\top)^2 + \frac{\lambda}{2} \left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 \right)$$
$$+ \frac{\beta}{2} \sum_{i=1}^{m} \sum_{j \in \mathcal{F}(i)} \mathbf{S}_{ij} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2. \tag{8}$$

Note that we leave out the superscript t for simplicity of notations. From this equation we can see that when $\beta = 0$, we recover SLOMA. Equation (8) can be solved by gradient descend methods [2], [3].

IV. Experiment

In this section, we introduce the details of the experiments, including the datasets, the evaluation metrics, and the baselines.

A. Datasets

We conduct the experiments on two real-world datasets: Yelp and Douban. Yelp is a location-based website where users can give ratings to and write reviews on items like restaurants, theaters, and businesses. The Yelp dataset is provided by the Yelp Dataset Challenge,[5] which has been used in previous research in recommendation [5], [18], [30]. Douban is a Chinese website where users can rate and share their opinions on items such as movies and books. The Douban Dataset is obtained from [15]. In both websites, users can build Facebook-style connections to each other and all ratings are in the range 1 to 5. Therefore, the datasets are ideal for evaluating the effectiveness of our proposel model.

---
[5]https://www.yelp.com/dataset_challenge

We preprocess the two datasets by removing users without any friends or with ratings fewer than 5. These users are called, respectively, "social cold-start" and "cold start" users in RSs. The statistics of the two preprocessed datasets are shown in Table I. Note that in the table, R_density represents the density of the rating matrix, and S_edges and S_density represent the social connections and density of the social matrix, respectively.

B. Evaluation Metrics

We choose two evaluation metrics, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). They are defined as

$$\text{MAE} = \frac{1}{|\bar{\Omega}|} \sum_{(i,j) \in \bar{\Omega}} |\mathbf{O}_{ij} - \mathbf{R}_{ij}|,$$

$$\text{RMSE} = \sqrt{\frac{1}{|\bar{\Omega}|} \sum_{(i,j) \in \bar{\Omega}} (\mathbf{O}_{ij} - \mathbf{R}_{ij})^2},$$

where $\bar{\Omega}$ is the set of all user-item pairs (i,j) in the test set, and $\mathbf{R}_{ij}$ is the corresponding rating predicted by the algorithm. These metrics are popular for the task of rating prediction in the literature [2], [3], [15].

C. Experimental Settings

We compare our proposed models with the following state-of-the-art methods:

- **RegSVD** [1]: It is the standard matrix factorization method with ℓ_2 regularization. We implement it according to [1].
- **LLORMA** [5]: It is based on the local low-rank assumption, which SLOMA also assumes. We implement it according to [5].
- **SocReg** [15]: It is a state-of-the-art method that integrates social connections into MF by employing social connections as regularization terms. We implement it according to [15].
- **SLOMA**: This is our proposed model. It utilizes social connections among users to form social groups for local low-rank factorization.
- **SLOMA++**: This is the same as SLOMA except that it applies social regularization to the factorization of submatrices, as is shown in Equation (7).

Following the experimental settings in [5], we also randomly split each dataset into training and test data with a ratio of 8:2. In the training process, the training data are used to fit the model, and the test data are used to calculate the prediction errors of the models. We repeat each experiment five times by randomly splitting the datasets and report the average results.

V. Analysis

In this section, we present and compare the experimental results of our proposed SLOMA and SLOMA++ against the baselines. We try to answer the following questions:

(1). What is the recommendation performance of our proposed models comparing to the state-of-the-art methods?

	Users	Items	Ratings	R_density	S_edges	S_density
Yelp	76,220	79,257	1,352,762	0.022%	647,451	0.022%
Douban	103,054	57,908	15,129,113	0.254%	753,358	0.028%

TABLE II

PERFORMANCE OF DIFFERENT METHODS IN YELP AND DOUBAN WITH $K = 10, 20$. THE BEST TWO RESULTS ARE HIGHLIGHTED. IMPROVEMENTS ARE MEASURED BY THE REDUCTION OF RMSE COMPARING TO SLOMA++.

Datasets	K	Metrics	RegSVD	LLORMA	SocReg	SLOMA	SLOMA++
Yelp	10	MAE	0.9478	0.9459	**0.9228**	0.9362	0.9301
		Improve	+1.87%	+1.67%	-0.79%	+0.65%	
		RMSE	1.1908	1.1843	1.1802	1.1760	**1.1755**
		Improve	+1.28%	+0.74%	+0.40%	+0.04%	
	20	MAE	0.9499	0.9477	0.9190	0.9389	0.9240
		Improve	+2.73%	+2.50%	-0.54%	+1.59%	
		RMSE	1.1918	1.1862	1.1754	1.1788	**1.1698**
		Improve	+1.85%	+1.38%	+0.48%	+0.76%	
Douban	10	MAE	0.5828	0.5811	0.5662	0.5744	**0.5603**
		Improve	+3.86%	+3.58%	+1.04%	+2.45%	
		RMSE	0.7347	0.7310	0.7165	0.7255	**0.7105**
		Improve	+3.29%	+2.80%	+0.84%	+2.07%	
	20	MAE	0.5803	0.5779	0.5638	0.5715	**0.5573**
		Improve	+3.96%	+3.56%	+1.15%	+2.48%	
		RMSE	0.7320	0.7278	0.7142	0.7225	**0.7080**
		Improve	+3.28%	+2.72%	+0.87%	+2.01%	

(2). How do the parameters in our models affect the overall performance? Specifically, how do the number of submatrices and hops, i.e., distance threshold, affect recommendation performance?

(3). What is the performance of different methods of connector selection?

(4). Which submatrix construction approach, heuristic or systematic, is better for SLOMA and SLOMA++?

A. Recommending Performance

To answer Question (1), we show the recommendation performance of the baselines and our proposed methods in Table II. We can see that SLOMA++ consistently outperforms all the other methods under different Ks in both datasets. This demonstrates the efficacy of integrating social connections with local low-rank framework. When comparing RegSVD, LLORMA, and SLOMA, we observe that SLOMA has better performance than LLORMA and RegSVD, which can be attributed to the better submatrices SLOMA builds with the help of social connections. When further studying the results of SocReg, SLOMA and SLOMA++, we can see that SLOMA++ is the best. This demonstrates that SLOMA++ makes the best use of social connections. Taking advantage from both the local low-rank assumption and social recommendation, SLOMA++ beats SocReg with the local low-rank framework and SLOMA with social regularization. Through these two comparisons, we can conclude that both social regularization and local low-rank can help improve the recommending performance.

When comparing SLOMA and SocReg, we can see that the performance of SocReg is consistently a bit better than that of SLOMA. This result not only indicates the power of social regularization but also points to a limitation of SLOMA, that is, it does not cover the long-tail users very well when constructing the submatrices and hence can only predict the long-tail users' ratings based on the mean of all the existing ratings. The latter also explains the phenomenon that the performance gain of SLOMA against RegSVD is limited. Therefore, it is very important to design better submatrix construction methods with larger coverage. We leave this for future research.

When comparing the performance on Yelp and Douban, we can see that the improvement on Douban is more significant than that on Yelp, which may be attributed to Douban's larger density of social connections (see Table I). It means that better performance can be obtained when social networks are dense. This is intuitive because the more friends a user has, the more likely his or her behaviors are affected by friends.

B. Impact of the Number of Local Models

In this section, we show how performance varies with different numbers of local models in LLORMA, SLOMA, SLOMA++. The results are shown in Figure 2. We also plot the RMSEs of RegSVD and SocReg for comparison. From the figures, we can see that with increasing number of local models, RMSE's of LLORMA, SLOMA, and SLOMA++ all decrease. This means that the more local models, the better performance we can get. However, when the number is large enough, e.g., around 30, the performance gains become marginal. This trend is consistent with the results reported in [5], which indicates that SLOMA and SLOMA++ show similar performance trends to LLORMA with the number of local models varying.

When the number of local models is smaller than 10, the performance is not as good as what we can achieve at 30. The reason is that fewer submatrices means less overlaps of the local models, which impairs the ensemble prediction accuracy according to Equation (6). Further, it also reduces the overall coverage of SLOMA and SLOMA++, which means more users' ratings are predicted by the mean of other users instead of by the learned model.

Again we can see that SLOMA++ consistently outperforms all the other baselines when the local model is large enough, e.g., greater than 10. In practice, it is good enough to set the number of local models to 50, which provides good prediction ability while avoiding higher costs of space and computation.
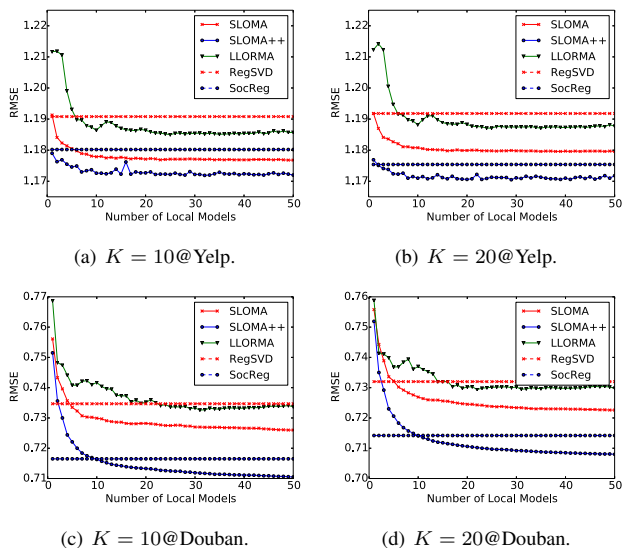

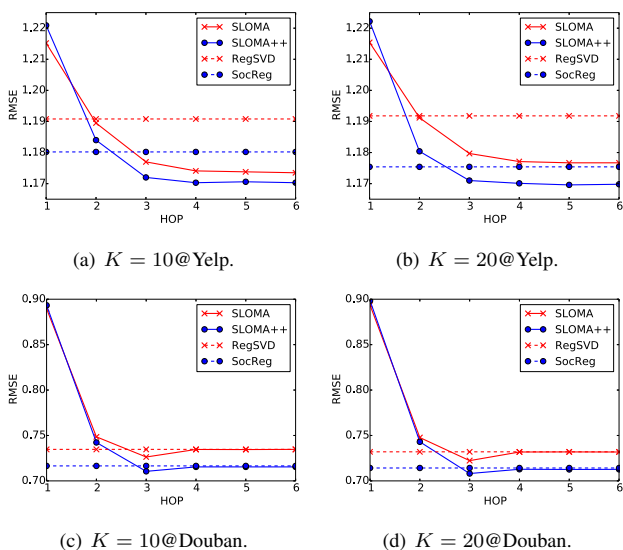Fig. 2. RMSEs vs different numbers of local models on Yelp and Douban.

Fig. 3. RMSEs vs different hops on Yelp and Douban.

C. Impact of the Number of Hops

In SLOMA and SLOMA++, we firstly select a small number of connectors, and then take all of their friends within different hops, i.e., the distance threshold d in Algorithm 2, in the social network to construct the social groups. Therefore, different hops mean different numbers of users included in each social group, thus different coverages of each local model, which results in different prediction capabilities of the proposed models. Due to the small-world phenomenon in social networks [31], [32], which states that two people can be connected by a small number of intermediates in social networks, the so-called Six Degree Separation. Thus, we limit the hop to the range $[1, 6]$ in the experiments.

Results are shown in Figure 3. As can be seen, the trend of the performance w.r.t. the hop is very similar to that of the number of local models, which is, the performance becomes better with increasing of hops and then becomes stable when the hop is large enough. Specifically, when $hop < 3$, the performance of SLOMA and SLOMA++ is worse than those of RegSVD and SocReg. The reason is that the coverage of each local model is not large enough, leading to a decrease in overall coverage of all local models. Therefore, the ensemble ability is impaired. However, when $hop \geq 3$, the performance of SLOMA and SLOMA++ becomes better, and stable when hop keeps increasing. Therefore, it means that a larger hop is useful for SLOMA and SLOMA++. However, it does not have to be very large because it will lead to higher space and computation cost but only marginal performance gain.

An interesting observation is that on Douban, at $hop > 3$, the performance of SLOMA and SLOMA++ decreases to a level that is very close to those of RegSVD and SocReg. This means that the benefit of the local low-rank assumption vanishes. We analyze the experimental results and find that SLOMA and SLOMA++ can cover all the users in Douban, thus each local model is reduced to the same one as RegSVD. Specifically, the variance of each local model decreases when the hop is too large, thus impairing the ensemble performance of all local models. Therefore, in the previous experiments comparing SLOMA and SLOMA++ against other baselines, we set $d = 3$ when constructing submatrices according to Algorithm 2.

In summary, Question (2) can be understood clearly based on the above two sections.

D. Impact of Connector Selection Methods

In this section, we try to answer Question (3) by exploring in depth the performance of different connector selection methods (introduced in Section III-E). When adopting the heuristic approaches to construct submatrices, the selection of connectors is very important. It determines the coverages of all local models. As we have elaborated, even though there exist some long-tail users who are rarely selected into any submatrix, we still want to design better heuristic approaches to include as many of them as we can. In Figure 4, we give the comparisons among different selection methods. Note that
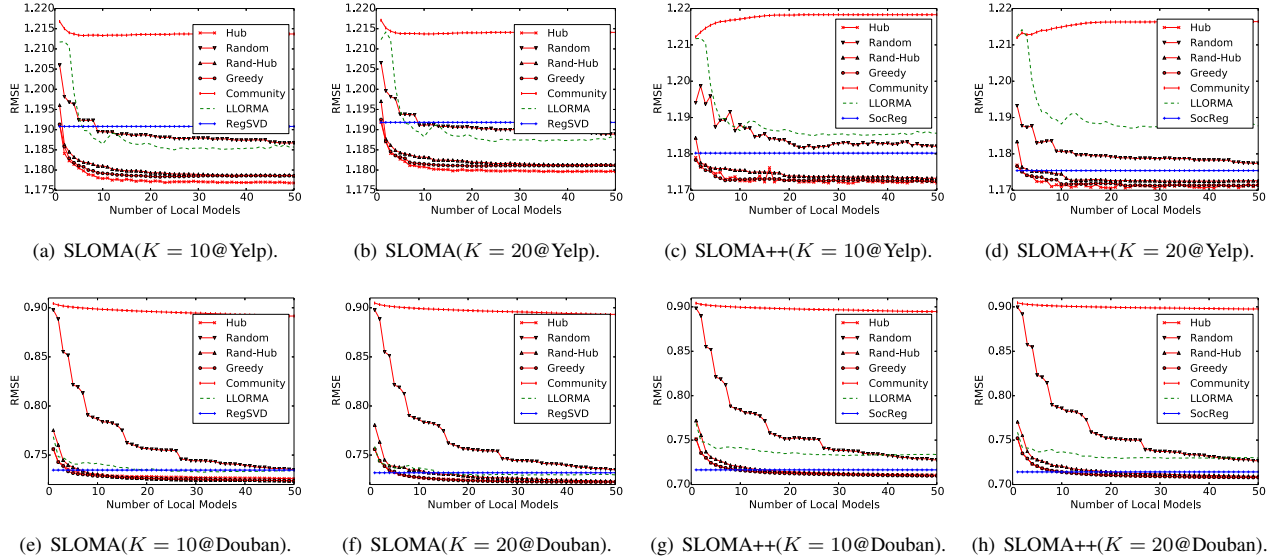
Fig. 4. RMSEs v.s different methods of identifying connectors on Yelp and Douban.

(a) SLOMA($K = 10$@Yelp). (b) SLOMA($K = 20$@Yelp). (c) SLOMA++($K = 10$@Yelp). (d) SLOMA++($K = 20$@Yelp).

(e) SLOMA($K = 10$@Douban). (f) SLOMA($K = 20$@Douban). (g) SLOMA++($K = 10$@Douban). (h) SLOMA++($K = 20$@Douban).

we also show the results of LLORMA as well as community-based SLOMA, and the results of RegSVD and SocReg are added for SLOMA and SLOMA++, respectively. We set $d = 3$ for different connector selection methods.

From Figure 4, we can see that Hub and Greedy outperform all the other methods with increasing number of local models. The better performance of Greedy lies in the fact that we choose uncovered users as connectors with the most neighbors each time, thus we can guarantee that all the connectors will not be too close to each other in the social graph. It will increase the coverage as well as the variance of all the submatrices, which is useful for the ensemble effect of all the local models. However, considering the clarity and simplicity of Hub, we adopt Hub as our selection method in the experiments reported in Section V-A. In Hub, each connector can be regarded as the most active and influential users of the social groups around them, which leads to a low-rank property of the constructed submatrix. Furhter, Hub is simple to implement.

We can see that for all the heuristic approaches, even though Random does not provide good performance, the RMSEs of them decrease with increasing number of local models. The underlying reason is that we can cover more entries in the rating matrix with more local models, leading to the performance gain for overall prediction. Taking all the heuristic approaches and LLORMA into consideration, it further demonstrates the efficacy of local low-rank framework.

E. Discussion on Systematic Submatrices Construction

From Figure 4, an obvious observation is that the community-based method, i.e., systematic submatrix construction, performs very bad in all settings. We make use of the BIGCLAM [26] to detect overlapping communities, each of which corresponds to one submatrix in SLOMA and SLOMA++. The number of communities is set to 50. However,

there are two problems facing the communities detected by BIGCLAM. First, BIGCLAM tends to detect closely connected groups, which leads to smaller coverage of each submatrix and fewer overlaps amongst all the submatrices. As we mentioned above, it impairs the ensemble effects in the overall prediction. Second, there are a small portion of long-tail users (it is around 4% in our experiments) who cannot be covered by all the communities, thus impairing the performance further. Therefore, it solves Question (4) as well as explains why we do not report the results of systematic approaches for SLOMA and SLOMA++ in Section V-A. However, we leave it as future work to detect better overlapping communities for SLOMA and SLOMA++.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we propose the SLOMA framework, which addresses the problem of submatrix construction facing LLORMA. To the best of our knowledge, it is the first work to incorporate social connections into the local low-rank framework. Based on the social homophily theory, we exploit users' social connections to construct meaningful as well as better submatrices, which leads to superior prediction model compared to LLORMA. Moreover, we integrate social regularization to further strengthen SLOMA. Extensive experiments have been conducted on two real-world datasets, comparing our models against MF as well as LLORMA. The results demonstrate that with the help of social connections, SLOMA outperforms LLORMA and MF.

VII. ACKNOWLEDGMENT

REFERENCES

[1] A. Paterek, "Improving regularized singular value decomposition for collaborative filtering," in *KDDcup Workshop*, vol. 2007, 2007, pp. 5–8.

[2] A. Mnih and R. Salakhutdinov, "Probabilistic matrix factorization," in *Advances in Neural Information Processing Systems*, 2007, pp. 1257–1264.

[3] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 426–434.

[4] Q. Yao and J. Kwok, "Accelerated inexact Soft-Impute for fast large-scale matrix completion," in *International Joint Conference on Artificial Intelligence*. AAAI Press, 2015, pp. 4002–4008.

[5] J. Lee and S. Kim, "LLORMA: Local low-rank matrix approximation," *Journal of Machine Learning Research*, vol. 17, no. 15, pp. 1–24, 2016.

[6] C. Chen, D. Li, Y. Zhao, Q. Lv, and L. Shang, "WEMAREC: Accurate and scalable recommendation through weighted and ensemble matrix approximation," in *International Conference on Research and Development in Information Retrieval*, 2015, pp. 303–312.

[7] C. Chen, D. Li, Q. Lv, J. Yan, S. Chu, and L. Shang, "MPMA: Mixture probabilistic matrix approximation for collaborative filtering," in *International Joint Conference on Artificial Intelligence*, 2016, pp. 1382–1388.

[8] M. Zhang, B. Hu, C. Shi, and B. Wang, "Local low-rank matrix approximation with preference selection of anchor points," in *International Conference on World Wide Web Companion*, 2017, pp. 1395–1403.

[9] Q. Yao and J. Kwok, "Colorization by patch-based local low-rank matrix completion," in *AAAI Conference on Artificial Intelligence*, 2015, pp. 1959–1965.

[10] X. Li, B. Shen, B. Liu, and Y. Zhang, "A locality sensitive low-rank model for image tag completion," *IEEE Transactions on Multimedia*, vol. 18, no. 3, pp. 474–483, 2016.

[11] K. Bhatia, H. Jain, P. Kar, M. Varma, and P. Jain, "Sparse local embeddings for extreme multi-label classification," in *Advances in Neural Information Processing Systems*, 2015, pp. 730–738.

[12] S. Sangho, C. Jaegul, J. Lee, and K. R. Chandan, "L-EnsNMF: Boosted local topic discovery via ensemble of nonnegative matrix factorization," in *International Conference on Data Mining*, 2016, pp. 479–488.

[13] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *International Conference on Information and Knowledge Management*, 2008, pp. 931–940.

[14] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *ACM Conference on Recommender Systems*, 2010, pp. 135–142.

[15] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *International Conference on Web Search and Data Mining*, 2011, pp. 287–296.

[16] J. Tang, X. Hu, H. Gao, and H. Liu, "Exploiting local and global social context for recommendation," in *International Joint Conference on Artificial Intelligence*, 2013, pp. 264–269.

[17] G. Guo, J. Zhang, and N. Yorke-Smith, "TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings." in *AAAI Conference on Artificial Intelligence*, 2015, pp. 123–129.

[18] H. Li, D. Wu, W. Tang, and N. Mamoulis, "Overlapping community regularization for rating prediction in social recommender systems," in *ACM Conference on Recommender Systems*, 2015, pp. 27–34.

[19] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[20] M. Girvan and M. Newman, "Community structure in social and biological networks," in *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, 2002, pp. 7821–7826.

[21] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.

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

[23] S. Zhang, R.-S. Wang, and X.-S. Zhang, "Identification of overlapping community structure in complex networks using fuzzy c-means clustering," *Physica A: Statistical Mechanics and its Applications*, vol. 374, no. 1, pp. 483–490, 2007.

[24] B. Rees and K. Gallagher, "Overlapping community detection by collective friendship group inference," in *International Conference on Advances in Social Network Analysis and Mining*, 2010, pp. 375–379.

[25] M. Coscia, G. Rossetti, F. Giannotti, and D. Pedreschi, "Demon: a local-first discovery method for overlapping communities," in *International Conference on Knowledge Discovery and Data Mining*, 2012, pp. 615–623.

[26] J. Yang and J. Leskovec, "Overlapping community detection at scale: a nonnegative matrix factorization approach," in *International Conference on Web Search and Data Mining*, 2013, pp. 587–596.

[27] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Computing Surveys*, vol. 45, no. 4, p. 43, 2013.

[28] J. Whang, D. F. Gleich, and I. S. Dhillon, "Overlapping community detection using seed set expansion," in *International Conference on Information and Knowledge Management*, 2013, pp. 2099–2108.

[29] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 43–52.

[30] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *ACM Conference on Recommender Systems*, 2013, pp. 165–172.

[31] J. Travers and S. Milgram, "The small world problem," *Phychology Today*, vol. 1, pp. 61–67, 1967.

[32] J. Kleinberg, "The small-world phenomenon: An algorithmic perspective," in *ACM Symposium on Theory of Computing*, 2000, pp. 163–170.

APPENDIX A
PROOF OF LEMMA III.1

Proof. Let $P_\Omega(A) = A_{ij}$ if $(i, j) \in \Omega$, and 0 otherwise. Then, we can express Equation (2) as

$$\min_{\mathbf{U}, \mathbf{V}} \frac{1}{2} \left\| P_\Omega(\mathbf{U}\mathbf{V}^\top - \mathbf{O}) \right\|_F^2. \tag{9}$$

If $\mathbf{U}$ and $\mathbf{V}$ are the optimal solution to Equation (9), then they should satisfy first order optimal condition as

$$\mathbf{U}^\top P_\Omega(\mathbf{U}\mathbf{V}^\top - \mathbf{O}) = 0, \tag{10}$$
$$P_\Omega(\mathbf{U}\mathbf{V}^\top - \mathbf{O})\mathbf{V} = 0.$$

As $\hat{\mathbf{U}} = \mathbf{U}\mathbf{Q}$ and $\hat{\mathbf{V}} = \mathbf{V}\mathbf{Q}^{-1}$, we have

$$\hat{\mathbf{U}}^\top P_\Omega(\hat{\mathbf{U}}\hat{\mathbf{V}}^\top - \mathbf{O}) = \mathbf{Q}^\top \mathbf{U}^\top P_\Omega(\hat{\mathbf{U}}\hat{\mathbf{V}}^\top - \mathbf{O})$$
$$= \mathbf{Q}^\top \mathbf{U}^\top P_\Omega(\mathbf{U}\mathbf{V}^\top - \mathbf{O}) = 0.$$

where the last equality comes from Equation (10). Similarly, we can get

$$P_\Omega(\hat{\mathbf{U}}\hat{\mathbf{V}}^\top - \mathbf{O})\hat{\mathbf{V}} = 0.$$

Thus, we can see $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ also satisfy the optimal condition of Equation (2), and then the Lemma holds. □