IJCNN 2019. International Joint Conference on Neural Networks. Budapest, Hungary. 14-19 July 2019

# Integrating Dual User Network Embedding with Matrix Factorization for Social Recommender Systems*

Liying Chen†, Honglei Zhang†, Jun Wu‡
*School of Computer and Information Technology*
*Beijing Jiaotong University*
Beijing 100044, China
{liyingchen, hongleizhang, wuj}@bjtu.edu.cn

***Abstract*—To address the data sparsity problem faced by recommender systems, social network among users is often utilized to complement rating data for improving the recommendation performance. One of current trends is to combine the idea of matrix factorization (MF) for predicting ratings with the idea of graph embedding (GE) for analyzing social network towards recommendation tasks. Despite enjoying many advantages, the existing integrated models have two critical limitations. First, such models are designed to work with either explicit or implicit social network, but little is known in taking both into account. Second, the users' embeddings learned by GE are fed to the downstream MF, but not reverse, which is sub-optimal because rating information is not considered for learning the users' embeddings. In this paper, we propose a novel social recommendation algorithm which exploits both explicit and implicit social networks towards the task of rating prediction. In Particular, we seamlessly integrate MF model and GE model within a unified optimization framework, in which MF and GE tasks can be reinforced each other during the learning process. Our encouraging experimental results on three real-world benchmarks validate the superiority of the proposed approach to state-of-the-art social recommendation methods.**

## I. Introduction

In the big data era, Recommender System (RS) has been an indispensable tool in helping us to filter the massive and evolving online information. The core technique behind RS is in modeling users' preference on items based on past user-item interactions, known as Collaborative Filtering (CF) [1]. Among various CF techniques, Matrix Factorization (MF) [2] might be the most popular one due to its excellent performance and high flexibility, as witnessed by the Netflix contest. The basic idea of MF is to embed both users and items into a shared latent vector space, and then the preference of users on items can be estimated by inner product between the user and item latent-factor vectors. One of the challenges faced by MF is the data sparsity problem, e.g., in the famous Netflix dataset, the rating density is less than 1% [3].

A promising solution to data sparsity is to incorporate the trust or social information into MF for facilitating recommendations based on the social correlation theory that people with close social relationship often have similar preference. Some studies along this line [4]–[8] utilize the social connections to constrain MF, so that the users with strong social correlation can be close to each other in the latent space, while others [9], [10] aim to learn a common users' representation by jointly factorizing the user-item rating matrix and the user-user social matrix. Most of such works [4]–[6], [9], [10] focus on exploiting the *explicit* social information (e.g., friend or trust relations among users), while only a few works [7], [8] leverage the *implicit* one which is inferred from user-item interactions.

In the meantime, a surge of efforts have been made in theories and algorithms for graph embedding (GE) [11], [12], including the traditional factorization-based methods, such as LLE [13], LE [14] and GF [15], and the prevailing neural network based methods, such as DeepWalk [16], node2vec [17] and DNGR [18]. We respectively term such two kinds of GE techniques as FGE and NGE in short. The former aims at factorizing a social matrix into two low-rank matrices, so that graph nodes can be embedded into a low-dimensional vector space; in essence, a few social recommendation techniques using joint MF model (e.g., [5], [10] and [6]) take the similar idea with FGE model to deal with social information. The latter is to learn the node embeddings with multiple levels of abstraction through neural networks; a few recent studies [8], [19] integrates MF model with NGE model and achieve better recommendation performance than traditional joint MF models.

Despite these successes, two base issues remain when we combine NGE model with MF model for the task of social recommendation. The first issue is how to choose the properties of social network which might be beneficial to the task of recommendation. Existing schemes of integrating NGE with MF [8], [19] train the NGE and MF models separately. It first trains a NGE model and then feeds the users' embeddings to the downstream MF part. Such a two-step approach only leverages the social information to improve the MF task, but not reverse. In this way, it is hard to determine which properties of social network should be preserved for recommendation

* This work was supported by the 'Natural Science Foundation of China (NSFC)' (61671048).
† These authors contributed equally to this work.
‡ To whom correspondence should be addressed.

tasks, as the rating information is not used in discovering the users' embeddings. Another tough issue is deciding to use explicit or implicit social network for recommendation. As demonstrated by [7], exploiting explicit social information achieves slightly better performance than using implicit one; in contrast, a recent study [8] argues that leveraging explicit social network may mislead the recommendation task. Until now, there is not a common view to determine which one is better yet.

To address the above issues, this paper investigates a novel framework of Matrix Factorization with Dual Graph Embedding, MFDGE in short, which seamlessly integrates MF model with NGE model by jointly optimizing a unified objective function. Specifically, MFDGE takes both explicit and implicit social networks into consideration towards the recommendation task. In contrast to the two-stage models such as [8], [19], our optimization scheme is able to achieve a much tighter coupling of both MF and NGE, where both MF and NGE tasks influence each other naturally and gradually via the joint optimization process. This interplay yields common representations for users learned from rating data in conjunction with explicit and implicit social information, which are more suitable for making accurate and reliable recommendations. To the best of our knowledge, our MFDGE scheme is the first dual graph embedding algorithm for solving social recommendation tasks with fully joint optimization of both MF model and NGE model. Our encouraging results from extensive experiments on three benchmarks show that exploiting dual social networks is better than using either one, and the proposed MFDGE scheme outperforms several state-of-the-art social recommendation methods.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 elaborates the proposed MFDGE approach. Extensive experiments are conducted in Section 4, followed by the conclusion in Section 5.

## II. Related Work

This paper investigates social recommendation by combining MF with GE, so we mainly review recent advance of MF-based and GE-based social recommendation algorithms. For comprehensive reviews of CF and GE, please refer to [1] and [12], respectively.

Existing MF-based social recommendation methods can be roughly categorized into two groups. In the first group, the social connections between users are taken as constraints, which are used to assist in factorizing the user-item ratings. For instance, Ma et al. [4] proposed a social trust ensemble method (RSTE) which linearly combines the basic MF model with a trust-based social neighborhood model. Jamali and Ester [5] proposed a SocialMF method that reformulates the contributions of trust users to the active user's representation. Ma et al. [6] proposed a SoReg method based on an assumption that if two users are socially connected, the learned low-dimensional representations should be also similar in the vector space, and applied such assumption as a regularization to MF. However, these methods are too heuristic and cannot improve the recommendation accuracy significantly.

The second group is to learn the common representations for users by jointly factorizing user-item ratings and user-user connections. For example, Ma et al. [9] proposed a SoRec method, the idea of which is to share a common user latent-feature matrix factorized by ratings and by trust. Guo et al. [10] proposed a TrustSVD method by extending the idea of the canonical SVD++ model [20], which takes into account both the explicit and implicit influence of ratings and trust information when predicting ratings of unknown items. The idea of exploiting social connections used by above methods is similar to the traditional factorization-based graph embedding techniques such as LLE [13], LE [14] and GF [15], however, which is inferior to the NGE solutions such as Deepwalk [16], node2vec [17] and DNGR [18], revealed by an experimental study [11] published recently.

The most related work to our approach is the CUNE method proposed by Zhang et al. [8] recently. It first extracted an implicit social network from the user-item ratings; then learned embedding for users from such network through a NGE model; and finally applied the embedding to assist in factorizing the user-item ratings. However, CUNE only uses implicit social network and holds the negative attitude towards explicit social network, leading to a single-faceted conclusion. Moreover, since the learning processes of MF and NGE are separated, it is hard to choose appropriate properties of social network for the task of social recommendation. This is exactly the main challenge we tackle in this paper.

## III. The Proposed MFDGE Approach

In this section, we start with some necessary preliminaries, and then illustrate the formulation of our MFDGE problem and the corresponding solution.

### A. Preliminaries

Suppose $\mathcal{U} = \{u_1, \cdots, u_m\}$ be a set with $m$ users, and $\mathcal{I} = \{i_1, \cdots, i_n\}$ denotes a set with $n$ items. The interactions associating $\mathcal{U}$ with $\mathcal{I}$ can be represented by a user-item rating matrix $\mathbf{R} \in [0,1]^{m\times n}$, where each element $r_{ui}$ indicates the rating of user $u$ rated on item $i$. Without loss of generality, we re-scale all ratings into the interval $[0,1]$, and adopt symbols $u, v$ to index users while $i, j$ to index items throughout this paper.

It's worth to note that, given a group of users $\mathcal{U}$, there exist explicit and implicit social connections simultaneously. So we define a *dual social network* on $\mathcal{U}$ as $\mathcal{G} =< \mathcal{U}, \mathcal{E}^E, \mathcal{E}^I >$, where $\mathcal{E}^E \in \{0,1\}^{m\times m}$ and $\mathcal{E}^I \in \mathbb{Z}^{m\times m}$ respectively record the explicit and the implicit connected relations among users. For the explicit user network, an edge $e^E_{uv} \in \{0,1\}$ is to indicate whether two users $u$ and $v$ are friend or not, which can be captured from the explicit social network directly. For the implicit user network, an edge $e^I_{uv} \in \mathbb{Z}$ denotes the implicit correlation between two users $u$ and $v$, which can be inferred from the user-item ratings through a technique named user-item bipartite network (U-I-Net) projection [8]. The basic idea

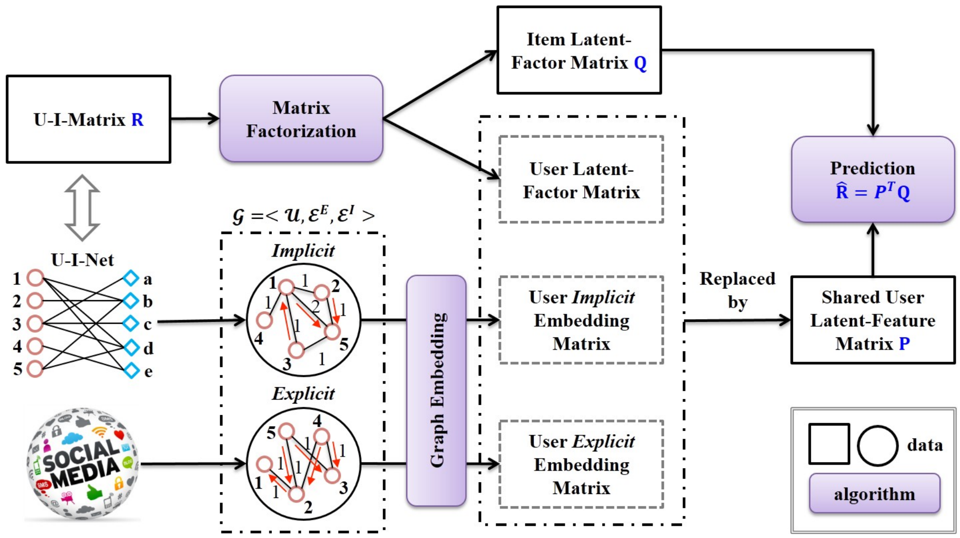



Fig. 1. The Schematic illustration of the MFDGE approach.

of U-I-Net projection is to connect two users who have rated at least one common item, and the corresponding edge weight denotes the number of the common items they rated. For example, $e^I_{uv} = 3$ means that user $u$ and $v$ have three common items they have rated in the past.

To exploit the complementary property between user-item ratings and user-user social connections, we will combine the MF model for predicting ratings with the NGE model for analyzing social connections through a unified learning framework in the next subsection.

## B. *Problem Formulation*

The framework of our MFDGE approach is illustrated by Fig. 1. At first, the UI rating matrix $\mathbf{R}$ is factorized into an item latent-factor matrix $\mathbf{Q} \in \mathbb{R}^{f\times n}$ and a user latent-factor matrix $\mathbf{P} \in \mathbb{R}^{f\times m}$; meanwhile, $\mathbf{R}$ is also represented as a U-I-Net which is then projected into an implicit user-user network; such an implicit network and an explicit one together form a dual social network $\mathcal{G} =< \mathcal{U}, \mathcal{E}^E, \mathcal{E}^I >$; next, by conducting NGE on $< \mathcal{U}, \mathcal{E}^E >$ and $< \mathcal{U}, \mathcal{E}^I >$ separately, two user embedding matrices could be gained; to merge MF and NGE models, we take user latent-factor matrix and two user embedding matrices as a common user latent-feature matrix between MF and NGE tasks, i.e. $\mathbf{P}$ is jointly learned from both $\mathbf{R}$ and $\mathcal{G} =< \mathcal{U}, \mathcal{E}^E, \mathcal{E}^I >$; finally, $\hat{\mathbf{R}} = \mathbf{P}^T\mathbf{Q}$ is expected to recover the observed ratings as well as predict the unobserved ones. Formally, the problem of our MFDGE is formulated as

$$\arg\min \mathcal{F} = \mathcal{L}_{MF} + \mathcal{L}_{NGE} + \mathcal{R} \tag{1}$$

where $\mathcal{L}_{MF}$, $\mathcal{L}_{NGE}$ and $\mathcal{R}$ denote the MF loss, NGE loss and regularization respectively.

**Matrix Factorization.** Suppose $\mathbf{p}_u$ and $\mathbf{q}_i$ are the user and item latent-factor vectors, a typical MF loss function is defined as a squared loss on observed ratings

$$\mathcal{L}_{MF}(\mathbf{p}_u, \mathbf{q}_i) = \sum_{(u,i)\in\Omega} \frac{1}{2}(r_{ui} - \hat{r}_{ui})^2 \tag{2}$$

where $\Omega$ is a set of the index of observed entries and $\hat{r}_{ui} = \mu + b_u + b_i + \mathbf{p}_u^T\mathbf{q}_i$ is the predicted rating of $r_{ui}$. For better modeling the preference of users on items, we introduce the biases of users and items for rating prediction. In details, $\mu$ denotes the overall average rating, $b_u$ and $b_i$ indicate the inherent character of user $u$ and the intrinsic quality of item $i$ respectively.

**Neural Dual-Graph Embedding.** A generalized NGE model consists of two phases, i.e. social corpus generation and node embedding. The former is to generate social corpus to identify the context users and negative users of each node, while the latter aims at performing node embedding to derive a low-dimensional representation for each node. We make some modifications to such two steps for the recommendation purpose.

*Step 1: Social Corpus Generation*

In the first step, we perform RandomWalk over a social network, and collect a set of node sequences that is called social corpus. Formally, given a user $u$, we apply a sliding window to generate the contexts $\mathcal{N}^+(u)$ by sliding on the generated sequences, where the middle position of the window is the center user $u$, and the other users who appear in the window are called contexts. Different from the regular RandomWalk, we additionally consider the number of common rated items $co(u, v)$ between user $u$ and user $v$ towards recommendation tasks, based on the assumption that if two users are socially connected and have common rated items, they will have a greater probability of becoming contexts in a social corpus. Here the jumping probability from $u$ to $v$ ($v \in \mathcal{S}_u$) is defined as:

$$p(v|u) = \frac{co(u, v)}{S} \tag{3}$$

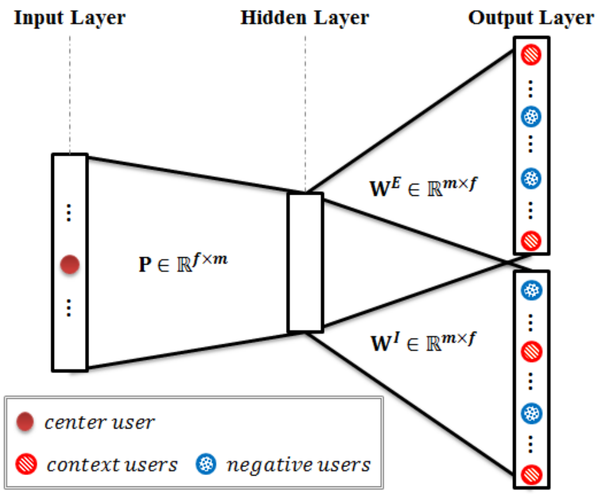



Fig. 2. The architecture of the modified SkipGram model.

where $S$ is a normalization factor, such that $S = \sum_{v' \in \mathcal{S}_u} co(u, v')$ and $\mathcal{S}_u \in \mathcal{U}$ is the set of friends linking user $u$.

On the other hand, for the generation of the negative samples set $\mathcal{N}^-(u)$, we apply a variant negative sampling technique to speed up the training process, due to the large number of users in social corpus. It takes into account not only the frequency of users in social corpus, but also the number of the users rated items in rating data. This is in line with the assumption that if a user has a high frequency in both social corpus and rating data, such user should have a high probability of becoming a negative sample. And the probability of being negative sample for user $u$ is defined as follows:

$$p(u) = \frac{g(u)^\gamma}{\sum_{u'} g(u')^\gamma} \tag{4}$$

where $g(u) = f(u) + r(u)$, and $f(u)$ denotes the frequency of the user $u$ in social corpus and $r(u)$ represents the number of user $u$ rated on items in rating data, here the parameter $\gamma$ is empirically set to 0.75.

Note that each node actually corresponds two groups of context users and negative users, which are generated from the explicit and the implicit social networks respectively.

*Step 2: Node Embedding via a Modified SkipGram*

SkipGram seeks to represent each user as a low-dimensional vector through a multi-layer perceptron (MLP) model, such that the users who are socially connected to each other will have similar vector representations. As mentioned before, since MF task and NGE task share the same user latent-feature representations in our scheme, $\mathbf{P} \in \mathbb{R}^{f\times m}$ can directly act as the weight matrix connecting input layer with hidden layer; moreover, since each user corresponds two groups of context users, we modify the model architecture of SkipGram as shown in Fig. 2. The center user is placed at the input layer; it is firstly transformed by $\mathbf{P}$, and then forwarded to two different output layers through two independent weight matrices $\mathbf{W}^E \in \mathbb{R}^{m\times f}$ and $\mathbf{W}^I \in \mathbb{R}^{m\times f}$; at the output layers, we obtain the predicted context users appearing in the explicit and implicit social networks respectively. Formally, the loss function of our NGE model with dual social network is defined as

$$\begin{aligned}\mathcal{L}_{NGE} &= \sum_u [-\log \sigma(\mathbf{p}_u^T \cdot \mathbf{h}^E) - \log \sigma(-\mathbf{p}_u^T \cdot \bar{\mathbf{h}}^E)] \\ &+ \sum_u [-\log \sigma(\mathbf{p}_u^T \cdot \mathbf{h}^I) - \log \sigma(-\mathbf{p}_u^T \cdot \bar{\mathbf{h}}^I)]\end{aligned} \tag{5}$$

with

$$\mathbf{h}^E = \frac{1}{s} \sum_{v \in \mathcal{N}_e^+(u)} \mathbf{w}_v^E, \tag{6}$$

$$\bar{\mathbf{h}}^E = \frac{1}{k} \sum_{v \in \mathcal{N}_e^-(u)} \mathbf{w}_v^E, \tag{7}$$

$$\mathbf{h}^I = \frac{1}{s} \sum_{v \in \mathcal{N}_i^+(u)} \mathbf{w}_v^I, \tag{8}$$

$$\bar{\mathbf{h}}^I = \frac{1}{k} \sum_{v \in \mathcal{N}_i^-(u)} \mathbf{w}_v^I, \tag{9}$$

where $\sigma$ is a sigmoid function; $\mathcal{N}_e^+(u)$ and $\mathcal{N}_i^+(u)$ respectively record the context users around $u$ derived from the explicit and implicit social networks; $\mathcal{N}_e^-(u)$ and $\mathcal{N}_i^-(u)$ are two negative sets of user $u$ sampled from the explicit and implicit social corpus respectively; and $s$ denotes the size of $\mathcal{N}_e^+(u)$ and $\mathcal{N}_i^+(u)$ while $k$ is the size of $\mathcal{N}_e^-(u)$ and $\mathcal{N}_i^-(u)$.

Note that $\mathbf{p}_u$ and $\mathbf{w}_u^E$ (or $\mathbf{w}_u^I$) are two representations of the user $u$, where $\mathbf{p}_u$ comes from columns of $\mathbf{P}$, which is the input$\rightarrow$hidden weight matrix, and $\mathbf{w}_u^E$ (or $\mathbf{w}_u^I$) comes from rows of $\mathbf{W}^E$ (or $\mathbf{W}^I$), which is the hidden$\rightarrow$output matrix. In subsequent analysis, we call $\mathbf{p}_u$ as the *input vector*, and $\mathbf{w}_u^E$ (or $\mathbf{w}_u^I$) as the explicit/implicit *output vector* of the user $u$. The intuition behind Eq. (5) is to maximize a function of inner product between the input vector of $u$ and the averaged output vector of the explicit/implicit context users around $u$, and minimize the same function between the input vector of $u$ and the averaged output vector of the negative examples that do not appear in the explicit/implicit context of $u$.

**Regularization**. To prevent over-fitting, we penalize the magnitude of variables by defining the regularization as

$$\begin{aligned}\mathcal{R} &= ||\mathbf{P}||_F^2 + ||\mathbf{Q}||_F^2 + ||\mathbf{W}^E||_F^2 + ||\mathbf{W}^I||_F^2 \\ &+ ||\mathbf{b}^{\mathcal{U}}||^2 + ||\mathbf{b}^{\mathcal{I}}||^2\end{aligned} \tag{10}$$

where $|| \bullet ||_F$ denotes the Frobenius norm of a matrix, $|| \bullet ||$ denotes the L2 norm of a vector, $\mathbf{b}^{\mathcal{U}} = [b_1, \cdots, b_m]^T$, and $\mathbf{b}^{\mathcal{I}} = [b_1, \cdots, b_n]^T$.

By putting everything together, the final objective of the

MFDGE problem is given by

$$\begin{aligned}\mathcal{L} &= \frac{1}{2}\sum_{(u,i)\in\Omega}(r_{ui}-\mu-b_u-b_i-\mathbf{p}_u^T\mathbf{q}_i)^2\\ &+ \alpha[-\log\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^E)-\log\sigma(-\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^E)]\\ &+ \beta[-\log\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^I)-\log\sigma(-\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^I)]\\ &+ \frac{\lambda_U}{2}||\mathbf{P}||_F^2+\frac{\lambda_I}{2}||\mathbf{Q}||_F^2\\ &+ \frac{\lambda_w}{2}(||\mathbf{W}^E||_F^2+||\mathbf{W}^I||_F^2)\\ &+ \frac{\lambda_b}{2}(||\mathbf{b}^{\mathcal{U}}||^2+||\mathbf{b}^{\mathcal{I}}||^2)\end{aligned}\tag{11}$$

where $\alpha$, $\beta$, $\lambda_U$, $\lambda_I$, $\lambda_w$ and $\lambda_b$ are hyper-parameters. By using a shared $\mathbf{P}$, both MF and NGE tasks can be reinforced each other. For one thing, under the supervision of user-item ratings, NGE will pay more attention on the properties of the dual social network which are helpful to the recommendation task. For another, by exploiting the higher-order proximity among users discovered from the dual social network, MF can learn more informative users' representations by taking both user-item ratings and user-user connections into account. Next, we will introduce the solution to our MFDGE problem.

**Algorithm 1** The MFDGE Algorithm.

**Input:**

$\mathbf{R}\in[0,1]^{m\times n}$: observed ratings;
$\mathcal{G}=<\mathcal{U},\mathcal{E}^E,\mathcal{E}^I>$: dual social network ;
$f$: embedding dimensions; $L$: the length of sequence;
$k$: number of negative samples; $s$: window size;
$\lambda_U$, $\lambda_I$, $\lambda_W$, $\lambda_b$, $\alpha$, $\beta$: trade-off parameters

```
Initialization P, Q, W^E, W^I, b^U, b^I;
Random walk on graphs < U, E^E > and < U, E^I > starting with every user to generate social corpus by Eq.( 3);
repeat
    Randomly select an index (u, i) from Ω;
    Calculate e_ui for user u on item i;
    Sliding window in the sequence consisting of user u to generate N_e^+(u) and N_i^+(u);
    Negative sampling in the sequence consisting of user u to generate N_e^-(u) and N_i^-(u) by Eq.( 4);
    Update p_u, q_i by Eq.( 12) and Eq.( 13);
    Update b_u^U and b_i^I by Eq.( 16) and Eq.( 17);
    for v ∈ N_e^+(u) ∪ N_e^-(u) do
        Update w_v^E using Eq.( 14);
    end for
    for v ∈ N_i^+(u) ∪ N_i^-(u) do
        Update w_v^I using Eq.( 15);
    end for
until converge
return P, Q, b^U, b^I;
```

*C. Solution*

We alternatively solve the six subproblems for MFDEG in Eq.(11): $\mathbf{P},\mathbf{Q},\mathbf{W}^E,\mathbf{W}^I,\mathbf{b}^{\mathcal{U}}$ and $\mathbf{b}^{\mathcal{I}}$ through a stochastic gradient descent (SGD) algorithm. The partial derivatives of Eq.(11) with respect to the six variables are

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\mathbf{p}_u} &= \sum_{i\in I_u}e_{ui}\cdot\mathbf{q}_i+\lambda_U\mathbf{p}_u\\ &+ \alpha[(\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^E)-1)\mathbf{h}^E+\sigma(\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^E)\bar{\mathbf{h}}^E]\\ &+ \beta[(\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^I)-1)\mathbf{h}^I+\sigma(\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^I)\bar{\mathbf{h}}^I]\end{aligned}\tag{12}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{q}_i} = \sum_{u\in U_i}e_{ui}\cdot\mathbf{p}_u+\lambda_I\mathbf{q}_i\tag{13}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_v^E}=\begin{cases}\frac{1}{s}[\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^E)-1]\cdot\mathbf{p}_u+\lambda_W\mathbf{w}_v^E, & if\ v\in\mathcal{N}_e^+(u)\\ \frac{1}{k}[\sigma(\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^E)]\cdot\mathbf{p}_u+\lambda_W\mathbf{w}_v^E, & if\ v\in\mathcal{N}_e^-(u)\end{cases}\tag{14}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_v^I}=\begin{cases}\frac{1}{s}[\sigma(\mathbf{p}_u^T\cdot\mathbf{h}^I)-1]\cdot\mathbf{p}_u+\lambda_W\mathbf{w}_v^I, & if\ v\in\mathcal{N}_i^+(u)\\ \frac{1}{k}[\sigma(\mathbf{p}_u^T\cdot\bar{\mathbf{h}}^I)]\cdot\mathbf{p}_u+\lambda_W\mathbf{w}_v^I, & if\ v\in\mathcal{N}_i^-(u)\end{cases}\tag{15}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{b}_u^{\mathcal{U}}} = \sum_{i\in I_u}e_{ui}+\lambda_b\mathbf{b}_u^{\mathcal{U}}\tag{16}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{b}_i^{\mathcal{I}}} = \sum_{u\in U_i}e_{ui}+\lambda_b\mathbf{b}_i^{\mathcal{I}}\tag{17}$$

where $e_{ui}=\hat{r}_{ui}-r_{ui}$ indicates the prediction error for user $u$ on item $i$, and $\mathcal{I}_u$ denotes the set of items rated by user $u$ while $\mathcal{U}_i$ is the set of users who rated item $i$.

Different from the regular neural network learning, which requires a back propagation (BP) mechanism to update the weight matrix connecting input layer with hidden layer (i.e. variable $\mathbf{P}$), in this work we share $\mathbf{P}$ between MF model and NGE model, and thus we can separately update $\mathbf{P}$ and $\mathbf{W}^E$ (or $\mathbf{W}^I$) in MF and NGE tasks. Since $\mathbf{W}^E$ (or $\mathbf{W}^I$) connects hidden layer with output layer, we can directly compute $\frac{\partial\mathcal{L}}{\partial\mathbf{w}_v^E}$ (or $\frac{\partial\mathcal{L}}{\partial\mathbf{w}_v^I}$). We summarize the solution for MFDGE in Algorithm 1.

## IV. Experiments

In this section, we first introduce our experimental settings, and then present the experimental results that validate the effectiveness of our approach. The experiments actually contain two parts. In the first part, we will compare our approach with several state-of-the-art methods for social recommendation. In the second part, we compare our algorithm with the MF model integrating different graph embedding solutions.

*A. Experimental Settings*

To verify the effectiveness of our method in terms of rating prediction, we utilize three real-world datasets respectively obtained from Filmtrust [1], Yelp [2] and Douban [3], all of which include both user-item ratings and user-user connections. The statistics about such datasets are summarized in Table I, Table II and Table III, in terms of ratings data, explicit social data and implicit social data respectively.

[1]http://www.librec.net/datasets.html
[2]https://www.yelp.com/dataset
[3]http://www.douban.com

TABLE I
STATISTICS OF THREE DATASETS ABOUT RATINGS DATA.

| Datasets | Users | Items | Ratings | Density |
|---|---|---|---|---|
| Filmtrust | 1,508 | 2,071 | 35,497 | 1.136% |
| Yelp | 3,993 | 6,443 | 230,781 | 0.897% |
| Douban | 2,963 | 39,694 | 894,887 | 0.760% |

TABLE II
STATISTICS OF THREE DATASETS ABOUT EXPLICIT SOCIAL DATA.

| Datasets | Followers | Followees | Links | Density |
|---|---|---|---|---|
| Filmtrust | 609 | 732 | 1,853 | 0.415% |
| Yelp | 3,516 | 3,516 | 136,722 | 1.105% |
| Douban | 2,745 | 2,741 | 35,771 | 0.475% |

We adopt two metrics to measure the recommendation accuracy, i.e. root mean square error (RMSE) and mean absolute error (MAE), defined as

$$RMSE = \sqrt{\frac{\sum_{(u,i)}(r_{ui} - \hat{r}_{ui})^2}{|\mathcal{T}|}}, \; MAE = \frac{\sum_{(u,i)} |r_{ui} - \hat{r}_{ui}|}{|\mathcal{T}|}$$

where $\mathcal{T}$ denotes the testing set and $(u, i) \in \mathcal{T}$, and $|\bullet|$ is the number of a set. The smaller RMSE and MAE values are, the better prediction performance is.

### B. Comparison with state-of-the-arts of Social Recommendation

In this part, we compare our MFDGE approach with several existing social recommendation approaches, including

- **MF [2]**: This is a canonical latent factor model which learns the latent representations for users and items in a common subspace by factorizing the user-item rating matrix. It serves as a baseline in the comparisons.
- **SVD++ [20]**: This is an improved variant of MF. Based on the MF framework, SVD++ further models the biases of users and items and takes the users' implicit feedback into consideration.
- **RSTE [4]**: This is an integrated model using both user-item ratings and user-user connections. For active users, RSTE not only models their personal preference on items but also takes their social neighbors' preference (on the same items) into consideration.
- **SocialMF [5]**: This is another integrated model, which argues that a user's preference is determined by his/her friends and SocialMF models the users' representations by the linear combination of their friends' representations.
- **SoReg [6]**: This is also an integrated model, which assumes that connected users should have similar latent representations and adopts a mechanism termed social regularization to extend MF model.
- **TrustSVD [10]**: This is an extended version of SVD++ by incorporates both rating implicit feedback and social implicit feedback into MF model.

TABLE III
STATISTICS OF THREE DATASETS ABOUT IMPLICIT SOCIAL DATA.

| Datasets | Followers | Followees | Links | Density |
|---|---|---|---|---|
| Filmtrust | 247 | 247 | 2,824 | 4.628% |
| Yelp | 3,401 | 3,401 | 153,058 | 1.323% |
| Douban | 692 | 692 | 41,696 | 8.707% |

- **CUNE [8]**: This is integrated model of MF and NGE, which conducts NGE on an implicit social network, which is extracted from user-item ratings, and then takes the embeddings as constraints to facilitate MF model.

All of above methods in conjunction with our MFDGE approach involve serval hyper-parameters. We tune all these parameters to their optimal values via 5-fold cross validation and grid search. For all the latent factor model, we set the embedding size as 10 for fair comparison. For our approach, we set the parameters $\alpha$ = $\beta$ =0.01 for Filmtrust and Yelp, and $\alpha$ = $\beta$ =0.0001 for Douban; $s$=40 for Filmtrust, $s$=10 for Yelp and $s$=30 for Douban; $k$=50 for Filmtrust, $k$=5 for Yelp and $k$=20 for Douban; $L$=20 for Filmtrust, Yelp and Douban; $\lambda_U$ =0.03, $\lambda_I$=0.01, $\lambda_W$=1, $\lambda_b$=0.01 for Filmtrust and Yelp, $\lambda_U$ =0.01, $\lambda_I$=0.02,$\lambda_W$=1,$\lambda_b$=0.01 for Douban.

Table IV illustrates the quantitative results in terms of RMSE and MAE performed by different methods on three datasets, where the best performance is boldfaced and the percentages indicate the relative improvement of our approach over other compared methods. Several observations can be drawn from the experimental results.

At first, all integrated models (RSTE, SocialMF, SoReg, TrustSVD, CUNE and MFDGE) outperforms the two baseline methods (MF and SVD++) that only use user-item ratings, which verifies the usefulness of social information in improving recommendation accuracy. Furthermore, by comparing the six social recommendation methods, the two NGE-based methods (CUNE and MFDGE) achieve better performance than others (RSTE, SocialMF, SoReg and TrustSVD), which validates the superiority of NGE to traditional heuristic methods using social information. Finally, by comparing CUNE and MFDGE, it is impressive that our MFDGE approach considerably outperforms CUNE. This observation reveals that integrating NGE and MF within a unified framework is essential. Such an integrated model can enforce NGE to choose the properties of the social network, which is helpful to recommendation.

### C. Comparison with different graph embedding solutions

We then compare the proposed MFDGE approach with the MF models that is integrated with different graph embedding solutions. We use LLE-MF, LE-MF, and GF-MF to denote the methods that integrates MF with Locally Linear Embedding (LLE), Laplacian Eigenmaps (LE), and Graph Factorization (GF), respectively. In order to inspect the difference between explicit and implicit social information, we include a variant of CUNE, termed CUNE+, in the comparison. The unique differ-

TABLE IV
PERFORMANCE COMPARISONS ON TWO REAL-WORLD DATASETS WITH STATE-OF-THE-ART RECOMMENDATIONS.

| Datasets | Metrics | MF | SVD++ | RSTE | SocialMF | SoReg | TrustSVD | CUNE | MFDGE |
|---|---|---|---|---|---|---|---|---|---|
| Filmtrust | RMSE | 0.87869 | 0.86800 | 0.84654 | 0.84782 | 0.83114 | 0.82879 | 0.82727 | **0.79457** |
| | Improve | 9.57% | 8.46% | 6.14% | 6.28% | 4.40% | 4.13% | 3.95% | |
| | MAE | 0.66427 | 0.64954 | 0.65970 | 0.65727 | 0.64120 | 0.63206 | 0.64334 | **0.60996** |
| | Improve | 8.18% | 6.09% | 7.54% | 7.20% | 4.87% | 3.50% | 5.19% | |
| Yelp | RMSE | 1.07944 | 1.05876 | 1.04474 | 1.04307 | 1.04857 | 1.04572 | 1.04145 | **1.02763** |
| | Improve | 4.80% | 2.94% | 1.64% | 1.48% | 2.00% | 1.73% | 1.33% | |
| | MAE | 0.85968 | 0.84302 | 0.82552 | 0.81919 | 0.81998 | 0.82221 | 0.81838 | **0.79551** |
| | Improve | 7.46% | 5.64% | 3.64% | 2.89% | 2.98% | 3.25% | 2.79% | |
| Douban | RMSE | 0.85645 | 0.84862 | 0.81357 | 0.79398 | 0.78006 | 0.79347 | 0.76739 | **0.74163** |
| | Improve | 13.41% | 12.61% | 8.84% | 6.59% | 4.93% | 6.53% | 3.36% | |
| | MAE | 0.66372 | 0.65794 | 0.63896 | 0.62561 | 0.61729 | 0.62555 | 0.6088 | **0.58426** |
| | Improve | 11.97% | 11.20% | 8.56% | 6.61% | 5.35% | 6.60% | 4.03% | |

TABLE V
PERFORMANCE COMPARISONS ON TWO REAL-WORLD DATASETS WITH DIFFERENT GRAPH EMBEDDING SOLUTIONS.

| Datasets | Metrics | LLE-MF | LE-MF | GF-MF | CUNE | CUNE+ | MFDGE |
|---|---|---|---|---|---|---|---|
| Filmtrust | RMSE | 0.84782 | 0.85260 | 0.83923 | 0.82727 | 0.81350 | **0.79457** |
| | Improve | 6.28% | 6.81% | 5.32% | 3.95% | 2.33% | |
| | MAE | 0.65727 | 0.65275 | 0.64388 | 0.64334 | 0.62684 | **0.60996** |
| | Improve | 7.20% | 6.56% | 5.27% | 5.19% | 2.69% | |
| Yelp | RMSE | 1.04307 | 1.09894 | 1.05367 | 1.04145 | 1.04045 | **1.02763** |
| | Improve | 1.48% | 6.49% | 2.47% | 1.33% | 1.23% | |
| | MAE | 0.81919 | 0.87782 | 0.83162 | 0.81838 | 0.81201 | **0.79551** |
| | Improve | 2.89% | 9.38% | 4.34% | 2.79% | 2.39% | |
| Douban | RMSE | 0.79398 | 0.78432 | 0.90601 | 0.76739 | 0.76652 | **0.74163** |
| | Improve | 6.59% | 5.44% | 18.14% | 3.36% | 3.25% | |
| | MAE | 0.62561 | 0.62201 | 0.69780 | 0.60880 | 0.60720 | **0.58426** |
| | Improve | 6.61% | 6.07% | 16.27% | 4.03% | 3.78% | |

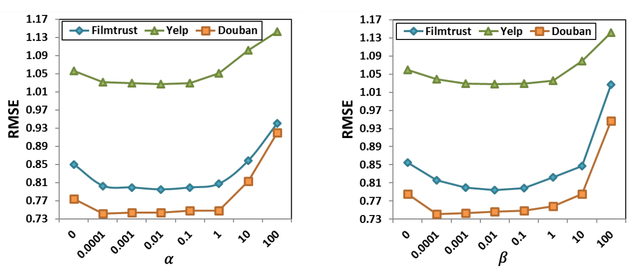


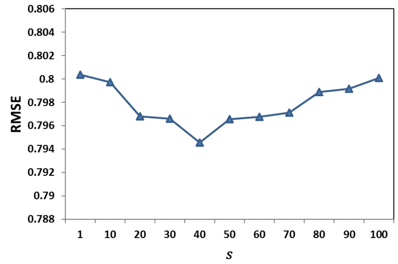

Fig. 3. Performance impact of parameter $\alpha$ and $\beta$ on three datasets.

ence between CUNE+ and CUNE is that the former performs NGE on the explicit social network while the latter uses the implicit network. We use the similar strategy introduced in the above subsection to tune the parameters of these methods for the fair comparison purpose.

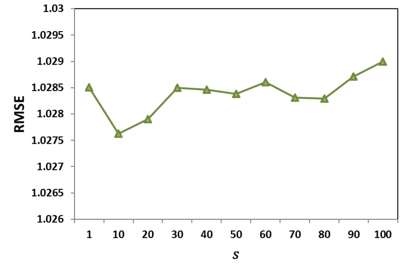
Table V presents the quantitative results in terms of RMSE and MAE of all compared methods on three datasets. From the experimental results, we can found that the three NGE-based approaches (CUNE+, CUNE and MFDGE) achieve better performance than the other three methods which exploit the social information either heuristically (LLE-MF and LE-MF) or using the traditional factorization-based graph embedding technique (GF-MF). This observation again demonstrates that NEG is superior to the traditional methods of exploiting social information for the task of recommendation. Moreover, CUNE+ and CUNE both utilize the single social network, and perform slight worse than MFDGE, which makes it clear that the explicit social network and the implicit one are mutually beneficial and we should utilize them together in social recommendation tasks. Last but not least, the performance of our MFDGE is always the best among the three NGE-based methods, which again demonstrates the benefit of integrating NGE with MF within a unified framework. That is, the motivation of this work is empirically verified.

### D. Impact of hyper-parameters

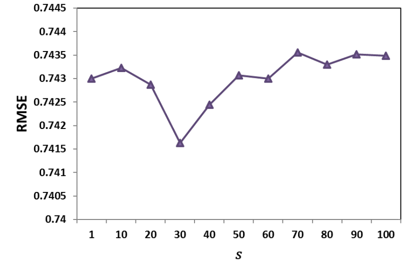
We investigate the impact of the key four parameters used by MFDGE, i.e. $\alpha$, $\beta$, $s$ and $k$. Fig. 3 shows the impacts of $\alpha$ and $\beta$ on three datasets. We observe that, with the increase of $\alpha$ and $\beta$ value, the curve drops first and then grows. It is not hard to explain that only exploiting rating data or social data cannot obtain satisfactory recommendation results, and the optimal proportion of each depends on the given dataset. As shown on the top of Fig. 4, it is not surprising that the performance curves of MFDGE trend to be worse when parameter $s$ exceeds a threshold. An intuitive explanation is that too higher-order proximity of nodes is not helpful to capturing the local structure. Furthermore, the impacts of parameter $k$ on three datasets in terms of RMSE are illustrated by the bottom of Fig. 4. With the growing of $k$ value, the RMSE decreases at



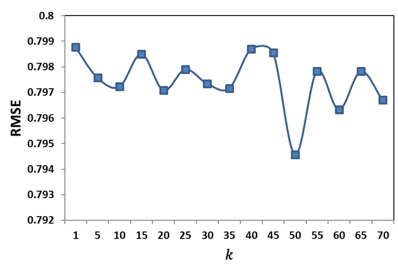

(a) Impact of $s$ on Filmtrust



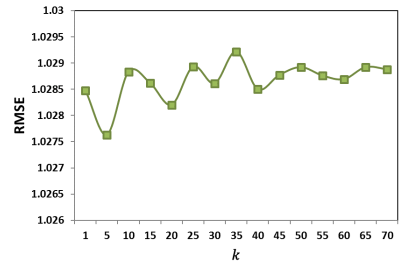

(b) Impact of $s$ on Yelp



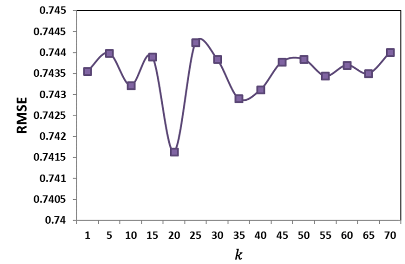

(c) Impact of $s$ on Douban




(d) Impact of $k$ on Filmtrust




(e) Impact of $k$ on Yelp




(f) Impact of $k$ on Douban

Fig. 4. Performance impact of parameter $s$ and $k$ on three datasets.

first and then increases. This phenomenon is consistent with the intuition that, if we use a large number of negatives, it is prone to introduce noise into the model training.

## V. Conclusion

In this paper, we have proposed a novel framework of Matrix Factorization with Dual Graph Embedding (MFDGE) for social recommender systems. In sharp contrast to existing integrated models of MF and GE, our MFDGE approach takes both explicit and implicit social networks into consideration, as well as performs a joint optimization of both MF and NGE tasks to achieve a tight coupling. Extensive experiments demonstrated that our MFDGE approach achieves better performance than state-of-the-art methods. As moving forward, we will exploit more auxiliary information concerning users and items [21] in the current framework to further enhance the latent representations for users and items. Furthermore, given a fact that the user network is time-variant, how to modify the existing time-variant graph analysis techniques [22], [23] for the purpose of social recommendation is another interested issue to be explored in the future.

## References

[1] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, p. 4, 2009.
[2] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.
[3] Y. Koren, "The bellkor solution to the netflix grand prize," *Netflix prize documentation*, vol. 81, pp. 1–10, 2009.
[4] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *SIGIR*, 2009, pp. 203–210.
[5] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *RecSys*, 2010, pp. 135–142.
[6] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *WSDM*, 2011, pp. 287–296.
[7] H. Ma, "An experimental study on implicit social recommendation," in *SIGIR*, 2013, pp. 73–82.
[8] C. Zhang, L. Yu, Y. Wang, C. Shah, and X. Zhang, "Collaborative user network embedding for social recommender systems," in *SDM*, 2017, pp. 381–389.
[9] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: Social recommendation using probabilistic matrix factorization," in *CIKM*, 2008, pp. 931–940.
[10] 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*, 2015, pp. 123–129.
[11] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Elsevier Knowl. Based Syst.*, vol. 151, pp. 78–94, 2017.
[12] H. Cai, V. W. Zheng, and K. Chang, "A comprehensive survey of graph embedding: problems, techniques and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 30, pp. 1616–1637, 2018.
[13] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
[14] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *NIPS*, 2002, pp. 585–591.
[15] A. Ahmed, N. Shervashidze, S. Narayanamurthy, V. Josifovski, and A. J. Smola, "Distributed large-scale natural graph factorization," in *WWW*, 2013, pp. 37–48.
[16] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *SIGKDD*, 2014, pp. 701–710.
[17] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *SIGKDD*, 2016, pp. 855–864.
[18] S. Cao, W. Lu, and Q. Xu, "Deep neural networks for learning graph representations." in *AAAI*, 2016, pp. 1145–1152.
[19] H. Zhang, G. Liu, and J. Wu, "Social collaborative filtering ensemble," in *PRICAI*, 2018, pp. 1005–1017.
[20] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *SIGKDD*, 2008, pp. 426–434.
[21] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Comput. Surv.*, vol. 47, no. 1, p. 3, 2014.
[22] H. Wang, W. Jia, X. Zhu, and C. Zhang, "Time-variant graph classification," *IEEE Trans. Syst., Man, Cybern.*, pp. 1–14, 2018.
[23] H. Wang, J. Wu, P. Zhang, and Y. Chen, "Learning shapelet patterns from networked time series," *IEEE Trans Ind. Informat.*, pp. 1–13, 2018.