Collaborative User Network Embedding for Social Recommender Systems

Chuxu Zhang* Lu Yu† Yan Wang* Chirag Shah* Xiangliang Zhang†

Abstract

To address the issue of data sparsity and cold-start in recommender system, social information (e.g., user-user trust links) has been introduced to complement rating data for improving the performances of traditional model-based recommendation techniques such as matrix factorization (MF) and Bayesian personalized ranking (BPR). Although effective, the utilization of the explicit user-user relationships extracted directly from such social information has three main limitations. First, it is difficult to obtain explicit and reliable social links. Only a small portion of users indicate explicitly their trusted friends in recommender systems. Second, the "cold-start" users are "cold" not only on rating but also on socializing. There is no significant amount of explicit social information that can be useful for "cold-start" users. Third, an active user can be socially connected with others who have different taste/preference. Direct usage of explicit social links may mislead recommendation. To address these issues, we propose to extract implicit and reliable social information from user feedbacks and identify top-k semantic friends for each user. We incorporate the top-k semantic friends information into MF and BPR frameworks to solve the problems of ratings prediction and items ranking, respectively. The experimental results on three real-world datasets show that our proposed approaches achieve better results than the state-of-the-art MF with explicit social links (with 3.0% improvement on RMSE), and social BPR (with 9.1% improvement on AUC).

Keywords: Social Recommender Systems, Network Embedding, Top-k Semantic Friends, Matrix Factorization, Bayesian Personalized Ranking

1 Introduction

Recommender systems have become an indispensable technique for filtering and recommending information for various needs. Various approaches [5, 6, 15, 16, 17, 18] based on low-rank matrix factorization (MF) or Bayesian personalized ranking (BPR) have been proposed to solve the problems of ratings prediction or items ranking. To improve the recommendation performance, recent works [1, 3, 20, 22] incorporate the observed explicit social information (e.g., trust connections of users) into MF or BPR frameworks to build novel systems (the so-called *social recommender systems*).

Social recommender techniques make usage of the user-user trustable social links to complement the sparse rating data, and thus improve the user preference prediction by considering not only a users' rating behavior, but also the preference of a user's trustable social neighbors. However, the additional information of explicit user-user trustable social links is not always available. In most real-life systems, such as Netflix or Ebay, there is no explicit indication about reliable social relationship. Even there is (e.g., in Epinions), the relationship indication is very sparse (e.g., the trust density in Epinions is 0.029%). Furthermore, the "cold-start" users are usually "cold" on both rating and on socializing. They seldom give ratings and hardly have explicit social connections to other users. Thus, explicit social information is very limited on helping the prediction of preference for "cold-start" users. Last but not least, active users often socially connect with many others, even with whom have different taste/preference. Direct usage of explicit social links without differentiation will mislead recommendation. In Section 2, we will present the observation of these limitations in three data sets that are popularly used in social recommender systems.

To resolve the above issues, we propose to extract *implicit* and *reliable* social information from user feedbacks (e.g., ratings or purchases) on items, and identify top-k semantic friends for each user. We design a novel CUNE (**C**ollaborative **U**ser **N**etwork **E**mbedding) method, which manages the feedbacks as a user-item bipartite network (U-I-Net) and then compresses the U-I-Net into a collaborative user network (C-U-Net) via one-mode projection onto users [23]. Next, inspired by the network embedding studies [2, 13, 19], we collect a set of node sequences (named "semantic social corpus") in C-U-Net via bias random walk. We utilize SkipGram model [12] to process the language composed of semantic social corpus. Finally, we compute the similarities of each two users in terms of their embedding vector rep-

*Rutgers University, USA. ({cz201, yw298}@cs.rutgers.edu, chirags@rutgers.edu)

†King Abdullah University of Science and Technology, SA. ({lu.yu, xiangliang.zhang}@kaust.edu.sa)

resentations returned by SkipGram. The k users having the largest similarities with a specific user are defined as the top-k semantic friends. CUNE also extends the DeepWalk [13] in generating random walks fed to SkipGram by: (a) only selecting unvisited neighbors of the predecessor node in the early stage, ensuring that each random walk can reach a large range; (b) choosing neighbors of the predecessor node with bias probabilities w.r.t. different connection weights, meaning the co-occurrence frequency of two nodes in the corpus reflects their correlation.

The top-k semantic friends better identify with whom a user has similar preference, comparing to the explicitly socially linked neighbors. We incorporate the information of top-k semantic friends into MF and BPR, following which we propose CUNE-MF (**CUNE** **M**atrix **F**actorization) and CUNE-BPR (**CUNE** **B**ayesian **P**ersonalized **R**anking) algorithms for the problems of ratings prediction and items ranking, respectively. In summary, our main contributions are as follows:

- We design the CUNE method to identify the top-k semantic friends of each user in recommender systems via one-mode projection and network embedding technique.
- We extends MF and BPR methods using the top-k semantic friends information generated by CUNE, and propose the CUNE-MF and CUNE-BPR algorithms for the problems of ratings prediction and items ranking, respectively.
- We conduct extensive experiments to evaluate the performances of CUNE-MF and CUNE-BPR. The results show that CUNE-MF outperforms the state-of-the-art MF with explicit social links (reducing prediction error by about 3.0%), and CUNE-BPR achieves much better results than the state-of-the-art social BPR (improving AUC score by about 9.1%).

The rest of the paper is organized as follows. Section 2 presents the analysis of explicit social information. Section 3 describes the proposed models and algorithms. The results from extensive experiments are presented in Section 4. Section 5 reviews related work, followed by the conclusion in Section 6.

2 Analysis of Explicit Social Information

In this section, we analyze the limitations of using explicit social information for building social recommender systems. The presented results are from three datasets, FilmTrust, CiaoDVD and Epinions, which are popularly used in social recommender system. An explicit link between two users indicates the trust relationship of them. The data statistics can be found in Table 1.

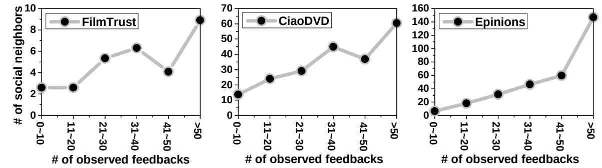

Figure 1: The social neighbors' sizes of different users grouped by the number of their feedbacks. The "cold-start" users have few neighbors.

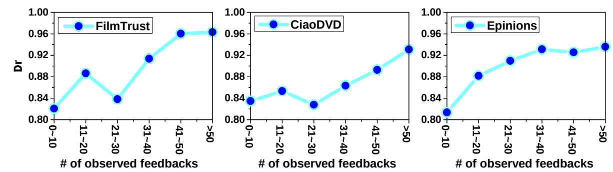

Figure 2: The Δr of different users grouped by the number of their feedbacks. Δr increases with the number of feedbacks.

The first straightforward observation is that users' reliable social relationship (e.g., trust connections) is difficulty to obtain. In most real-world situations, such as Netflix or Ebay, there is no explicit indication about reliable social relationship. Even there are, the indication matrix is extremely sparse. For example, in the three data sets under our analysis (FilmTrust, CiaoDVD, and Epinions), the trust densities are only around 0.42%, 0.23% and 0.029%, respectively.

Second, we investigate the "cold-start" users who have ratings on very few number of items, through analyzing how many explicit social links each user has, i.e., the size of the social neighborhood with explicit links. Figure 1 reports the average size of the social neighborhood of all users in different classes grouped by the number of their feedbacks (i.e., ratings). In all of the three datasets, we observe that the users with fewer feedbacks tend to have a smaller number of neighbors. Thus, for the "cold-start" users, the sparse social connections provide limited information for shaping their latent features as well as improving the recommendation results.

Third, we study whether active users, who have many feedbacks and social connections, have obvious benefits when adding social links in recommendation. There is an intuitive example. User 1 trusts an active user 2 (with a large number of feedbacks and social connections) due to their compatible movie preferences on love movie (e.g. having similar feedbacks on movies tagged with love or romance). However, user 1 and user 2 may have different tastes on other kinds of movies like action movies. Thus, the connection between user 1 and user 2 encodes a partial correlation that does not help predict user 2's feedbacks on action movie. Even worse, the connections between user 2 and other users who have partial common interests in different types of movies can mislead the preference inference of user 2. To formulate this issue, we define Δr as the difference of the feedbacks (i.e. ratings) on the same item given by one user and their social neighbors. The average results of Δr of all users in different classes grouped by the number of their feedbacks are shown in Figure 2. We can see that Δr increases with the increment of feedbacks number. Therefore, the explicit social connections between an active user and their social neighbors encode partial correlation. Such connections benefit little or even introduce noises for shaping active users' latent features.

In summary, the *explicit* social information is difficult to obtain, sparse to use and noisy to incorporate. We thus propose to extract *implicit* and *reliable* social information from user feedbacks that are easy to access. The extracted information is used to identify top-k semantic friends for each user and facilitate the recommendation.

3 Proposed Methods

In this section, we first introduce the CUNE method to identify the top-k semantic friends of each user, and then present the semantic social recommendation models using the top-k semantic friends information.

3.1 Top-k Semantic Friends Generation via Network Embedding

Our proposed CUNE method consists of three consecutive steps: 1) constructing collaborative user network via one-mode projection; 2) generating semantic social corpus by bias random walk; and 3) learning users' latent representations via SkipGram, to identify the top-k semantic friends of each user. Figure 3 is an illustration of these steps in CUNE method. Next, we introduce these steps one by one.

Step1: User-Item Bipartite Network Projection. At the first step of CUNE, we use one-mode projection to compress user-item bipartite network and construct collaborative user network. The user-item bipartite network (U-I-Net) represents the user-item feedbacks (e.g., ratings or purchases) by using two sets of nodes (i.e., users and items) and connecting two nodes from different sets. For example, edge e_{ij} denotes that user i has posted feedback on item j. We conduct one-mode projection onto users and build a user-only network where two users are connected if they share at least one neighboring item, such as the toy example shown in Figure 3 (step 1). Thereafter, we obtain a weighted collaborative user network (i.e., C-U-Net). In C-U-Net, an edge connects two users if they share at least a neighboring item in U-I-Net, and the edge weight denotes the number of their common neighbors. Besides, users who have indirect correlations in U-I-Net can be represented by hopping neighbors. For example, both user 1 and user 2 in Figure 3 are neighbors of item b, both user 1 and user 4 connect to item e. Thus edges e_{12} and e_{14} are weighted by 1. Moreover, transitive correlation is created between user 2 and 4 since they are 2-hop neighbors. Therefore, C-U-Net not only captures the correlations between neighboring users but also infers the transitive correlations among non-neighboring users.

Step2: RandomWalk with Bias. To conduct the second step of CUNE, we perform random walk samplings over C-U-Net, and collect a set of node sequences that are called the *semantic social corpus*. A random walk rooted at a specific node is a stochastic process with a sequence of random variables, such that each variable is a node chosen randomly from the neighbors of its predecessor. Different from the standard random walk, we select each successor node with bias probability to collect semantic social corpus fed to SkipGram. Formally, given a root node c_0, we simulate a bias random walk with length L in C-U-Net. Denote the i-th node in the walk as c_{i-1} and we select it from the neighbors of its predecessor c_{i-2} by two ways: (1) randomly select one neighbor if all neighbors of c_{i-2} have already been visited (selected in previous steps), (2) otherwise (some neighbors of c_{i-2} have not been visited), choose one of unvisited neighbors of c_{i-2} by $p(c_{i-1} = x|c_{i-2} = v) = \omega_{vx}/S$, where ω_{vx} is the edge weight between node v and x, and S is the normalization factor such that $S = \sum_{x\in \bar{N}(v)} \omega_{vx}$ ($\bar{N}(v)$ is the set of unvisited neighbor of v). We filter out the nodes which have no neighbors. The semantic social corpus meets the following expectations:

- The frequencies of nodes in the random walks follow a power-law-like distribution [13], similar to the distribution of word frequencies in text corpus. Thus, we are inspired to utilize SkipGram for learning feature embedding of each user in C-U-Net, which is analogous to Word2Vec embeddings.
- Each random walk not only creates correlations between neighboring nodes, but also considers transitive correlations among non-neighboring nodes. In addition, the co-occurrence frequencies of non-neighboring nodes in the same walks have an effect if we generate plenty of walks rooted at each node.
- In the early stage, we only choose unvisited neighbors of the predecessor so that each walk can reach a large range. Moreover, we select an unvisited neighbor

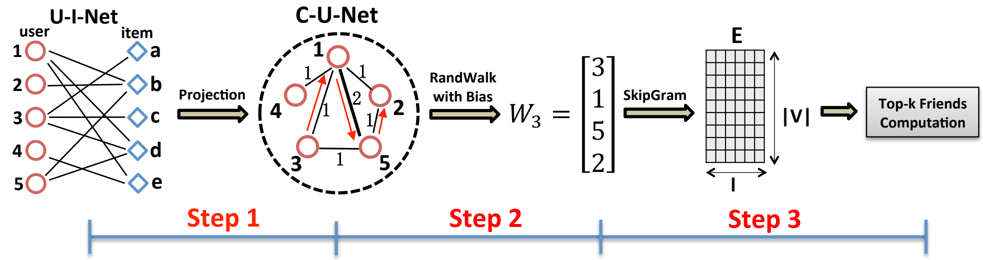

Figure 3: The illustration of CUNE method. The first step is to construct collaborative user network (C-U-Net) via one-mode projection onto users of user-item bipartite network (U-I-Net). The second step is to generate semantic social corpus (e.g., W_3) via random walk with bias. The third step is to learn users' latent representations (E) via SkipGram using semantic social corpus.

with probability w.r.t. its connection weight with the predecessor, therefore the co-occurrence frequency of two nodes in the corpus reflects their correlation.

Step3: Feature Embedding via SkipGram. SkipGram maximizes the average log probability of a specific word sequence appearing in a corpus. Formally, given a sequence of words $S_{\omega_0} = (w_0, w_2, \cdots, w_M)$, the objective function is:

$$maximize \ \frac{1}{M}\sum_{t=1}^{M}\sum_{-b\le t-t'\le b} log\ p(w_{t'}|w_t)$$

where b is the window size of context and $p(w_{t'}|w_t)$ is defined as Softmax function [12]. Similarly, we maximize the probability of node co-occurrences in each random walk W_v with fixed length L:

$$maximize \sum_{v_t\in W_v}\sum_{-\tau\le t-t'\le\tau} log\ p(v_{t'}|E(v_t))$$

where τ is the window size of v_t's context, i.e. $v_{t-\tau}\dots v_{t+\tau}$. Thus SkipGram learns a feature embedding E with $|V|\times l$ free parameters (V is the set of all nodes in C-U-Net) and each row of E denotes the feature vector (with size l) of a specific user. Figure 3 (step 3) shows a toy example: a walk $W_3 : 3 \to 1 \to 5 \to 2$ is generated by bias random walk (step 2) in C-U-Net. Then E is updated via SkipGram using W_3. In this work, the Hierarchical Softmax [12] is applied to approximate $p(v_{t'}|E(v_t))$ in order to avoid the complexity of computing the normalization factor in the Softmax function. We simulate T walks rooted at each node to generate the semantic social corpus and use stochastic gradient descent for training. After the embedding E is learned, we compute the cosine similarity for each pair of two users w.r.t. their embeddings and extract the top-k semantic friends.

3.2 Semantic Social Recommender Systems

We next introduce how to incorporate top-k semantic friends into low-rank matrix factorization (MF) for ratings prediction, and into Bayesian personalized ranking (BPR) for items ranking.

Ratings Prediction by CUNE-MF. The ratings of m users on n items is denoted by an $m\times n$ matrix R. MF approximates R by a multiplication of d-rank user and item latent vectors, i.e. $R \approx Z^TQ$, where $Z \in \mathbb{R}^{d\times m}$ and $Q \in \mathbb{R}^{d\times n}$ with $d \ll min(m, n)$. Thus the rating of user i on item j can be predicted by the inner product of the specific user latent feature $\boldsymbol{z}_i$ and item latent feature $\boldsymbol{q}_j$, i.e., $\hat{r}_{ij} = \boldsymbol{z}_i{}^T\boldsymbol{q}_j$. We incorporate the top-k semantic friends information generated by CUNE into MF as a regularization term and minimize the defined objective function of CUNE-MF:

$$\Phi \equiv \underbrace{\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(r_{ij}-\boldsymbol{z}_i{}^T\boldsymbol{q}_j)^2}_{loss\ function} + \underbrace{\frac{\lambda}{2}(\|Z\|_F^2+\|Q\|_F^2)}_{model\ regularizer}$$
$$+\underbrace{\frac{\alpha}{2}\sum_{i=1}^{m}\sum_{f\in H(i)}\|\boldsymbol{z}_i-\boldsymbol{z}_f\|_F^2}_{semantic\ social\ regularizer}$$

where I_{ij} is the indicator function which equals to 1 if user i rated item j and equals to 0 otherwise, and λ is the regularization parameter that controls model complexity to avoid over-fitting. $H(i)$ denotes the set of top-k semantic friends of user i and the parameter α controls the extent of the semantic social constraint. We tend to reduce the distance between the feature vectors of user i and their top-k semantic friends since smaller distance values mean closer tastes. When $\bar{H}(i)$ denotes the set of users whose top-k semantic friends set contains user i, we use the following gradient equations to perform the stochastic gradient descent for each observed rating r_{ij}:

$$\frac{\partial \Phi}{\partial \boldsymbol{z}_i} = (r_{ij} - {\boldsymbol{z}_i}^T \boldsymbol{q}_j)\boldsymbol{q}_j + \lambda \boldsymbol{z}_i + \alpha\Big[\sum_{f \in H(i)} (\boldsymbol{z}_i - \boldsymbol{z}_f) - \sum_{g \in \bar{H}(i)} (\boldsymbol{z}_g - \boldsymbol{z}_i)\Big]$$

$$\frac{\partial \Phi}{\partial \boldsymbol{q}_j} = (r_{ij} - {\boldsymbol{z}_i}^T \boldsymbol{q}_j)\boldsymbol{z}_i + \lambda \boldsymbol{q}_j$$

Items Ranking by CUNE-BPR. The preference feedbacks of m users on n items can be represented by an $m \times n$ boolean matrix G, where $g_{ui} = 1$ means user u posts positive feedback (e.g. "like" or purchase) on item i, while $g_{ui} = 0$ represents u's negative or unobserved feedback on item i (e.g. u is not interested in or not aware of item i). Denote x_{ui} and x_{uj} as u's preference scores on items i and j, respectively. Let P_u and N_u represent the positive and negative sets of items of u, respectively. BPR models users' preferences on different items and maximizes the following posterior probability over the parameter $\boldsymbol{\omega}$:

$$\prod_{u \in U} p(\boldsymbol{\omega}| >_u) \propto \prod_{u \in U} p(>_u |\boldsymbol{\omega}) p(\boldsymbol{\omega})$$

where U is the set of all users and notation $>_u \equiv \{x_{ui} > x_{uj}, i \in P_u, j \in N_u\}$ denotes the pairwise structure of u. Motivated by SBPR [22], we extend BPR by considering:

$$x_{ui} > x_{uk}, x_{uk} > x_{uj}, i \in P_u, k \in IP_u, j \in N_u$$

where IP_u denotes a set of items for which u did not post positive feedback, but at least one of their top-k semantic friends did. It is thus a natural assumption that user u likes i better than k, and likes k better than j. Thus we redefine the notation $>_u \equiv \{x_{ui} > x_{uk}, x_{uk} > x_{uj}, i \in P_u, k \in IP_u, j \in N_u\}$, which incorporates users' potential preference implied by their semantic friends. Then, the optimization criterion/likelihood for each user u is defined as:

$$\prod_{i,k \in T_u} \Big[p(x_{ui} > x_{uk})^{\delta(u,i,k)} (1 - p(x_{ui} > x_{uk}))^{1-\delta(u,i,k)}\Big]$$
$$\prod_{k,j \in \hat{T}_u} \Big[p(x_{uk} > x_{uj})^{\theta(u,k,j)} (1 - p(x_{uk} > x_{uj}))^{1-\theta(u,k,j)}\Big]$$

Here, $p(>_u |\boldsymbol{\omega})$ is written as $p(>_u)$ for simplicity. $T_u = P_u \cup IP_u$, $\hat{T}_u = IP_u \cup N_u$. $\delta(u, i, k)$ equals to 1 if $i \in P_u$ and $k \in IP_u$, and 0 otherwise. Similarly, $\theta(u, k, j)$ equals to 1 if $k \in IP_u$ and $j \in N_u$, and 0 otherwise. Due to the antisymmetric nature of pairwise structure, the posterior probability becomes:

$$\prod_{i \in P_u, k \in IP_u} p(x_{ui} > x_{uk}) p(\boldsymbol{\omega}) \prod_{k \in IP_u, j \in N_u} p(x_{uk} > x_{uj}) p(\boldsymbol{\omega})$$

Let $p(x_{ui} > x_{uk}) = \sigma(d(x_{ui}, x_{uk}))$, where σ is the sigmoid function: $\sigma(x) = \frac{1}{1+e^{-x}}$, and $d(x_{ui}, x_{uk})$ quantifies the difference between u's preference scores on i and k. For the explicit feedback, we define $d(x_{ui}, x_{uk}) = x_{ui} - x_{uk}$. For the semantic social feedback, we control the extent of semantic social influence by using a parameter s and define $d(x_{uk}, x_{uj}) = (x_{uk} - x_{uj})/s$. Similar to CUNE-MF, the preference function is modeled by matrix factorization, i.e. $x_{ui} = {\boldsymbol{z}_u}^T \boldsymbol{q}_i$ where $\boldsymbol{z}_u$ and $\boldsymbol{q}_i$ denote d-dimension features of user u and item i, respectively. Clearly, $\boldsymbol{\omega} \equiv (Z, Q)$ in our model, which can be initialized by prior $\mathcal{N}(\boldsymbol{0}, \lambda_\omega \boldsymbol{I})$ or small random value within [0,1]. By taking log-form of posterior probability, we define to maximize the objective function of CUNE-BPR as:

$$\varphi \equiv \sum_{u \in U} \Big[\underbrace{\sum_{i \in P_u} \sum_{k \in IP_u} ln\ \sigma(x_{ui} - x_{uk})}_{explicit\ feedback} + \underbrace{\sum_{k \in IP_u} \sum_{j \in N_u} ln\ \sigma(\frac{x_{uk} - x_{uj}}{s})}_{semantic\ social\ feedback} \Big] - \underbrace{\frac{\lambda_\omega}{2}(\|Z\|_F^2 + \|Q\|_F^2)}_{prior\ constraint}$$

Similar to CUNE-MF, we use the stochastic gradient descent for training. CUNE-BPR uses the following gradient equations to perform the parameters updating for each observed structure (u, i, k, j):

$$\frac{\partial \varphi}{\partial \boldsymbol{z}_u} = \frac{e^{-(x_{ui} - x_{uk})}}{1 + e^{-(x_{ui} - x_{uk})}}(\boldsymbol{q}_i - \boldsymbol{q}_k) + \frac{\frac{1}{s}e^{-(\frac{x_{uk} - x_{uj}}{s})}}{1 + e^{-(\frac{x_{uk} - x_{uj}}{s})}}(\boldsymbol{q}_k - \boldsymbol{q}_j) - \lambda_\omega \boldsymbol{z}_u$$

$$\frac{\partial \varphi}{\partial \boldsymbol{q}_i} = \frac{e^{-(x_{ui} - x_{uk})}}{1 + e^{-(x_{ui} - x_{uk})}}\boldsymbol{z}_u - \lambda_\omega \boldsymbol{q}_i$$

$$\frac{\partial \varphi}{\partial \boldsymbol{q}_k} = \frac{e^{-(x_{ui} - x_{uk})}}{1 + e^{-(x_{ui} - x_{uk})}}(-\boldsymbol{z}_u) + \frac{\frac{1}{s}e^{-(\frac{x_{uk} - x_{uj}}{s})}}{1 + e^{-(\frac{x_{uk} - x_{uj}}{s})}}\boldsymbol{z}_u - \lambda_\omega \boldsymbol{q}_k$$

$$\frac{\partial \varphi}{\partial \boldsymbol{q}_j} = \frac{\frac{1}{s}e^{-(\frac{x_{uk} - x_{uj}}{s})}}{1 + e^{-(\frac{x_{uk} - x_{uj}}{s})}}(-\boldsymbol{z}_u) - \lambda_\omega \boldsymbol{q}_j$$

4 Experiments

This section reports extensive evaluation results of our methods and other counterparts.

4.1 Experimental Design

Datasets. We use three datasets including FilmTrust[1], CiaoDVD[2] and Epinions[3]. All of them contain explicit social trust information and each trust value equals to 1.

[1] http://www.librec.net/datasets.html
[2] http://www.librec.net/datasets.html
[3] http://www.trustlet.org/epinions.html

Table 1: Statistics of datasets used in this paper.

Statistics	FilmTrust	CiaoDVD	Epinions
users #	1,508	17,615	40,163
items #	2,071	16,121	139,738
ratings #	35,497	72,665	664,824
density	1.14%	0.03%	0.01%
rating range	[0.5, 4]	[1, 5]	[1, 5]
trusts #	1,853	111,781	487,183
trust density	0.42%	0.23%	0.029%

The main statistics of these datasets are summarized in Table 1. In ratings prediction, we use ratings which are larger than a given threshold δ_r ($\delta_r = 3$) for constructing U-I-Net and C-U-Net since relative high ratings represent collaborative users' similar preferences for items. For the items ranking problem, we extract all ratings which are larger than δ_p ($\delta_p = 3$) and treat them as positive feedbacks.

Comparison Methods. We use two sets of comparison methods for two different tasks. For the task of ratings prediction, since our model builds on the top of regularization optimization framework, we compare it with related models with explicit or implicit social regularization constraint, as well as the traditional matrix factorization approach. Specifically, our CUNE-MF is compared with: **(a) NMF** [7]: a widely used non-negative low-rank matrix factorization technique. **(b) IMF** [9]: an extension of MF with the implicit social regularization terms computed by rating-based similarity function. **(c) EMF** [11]: a regularized MF via the explicit social connections. For the task of items ranking, we compare our CUNE-BPR method with the Bayesian personalized ranking model and the related method with the explicit social information, as well as a baseline solution. These methods are: **(1) MP**: ranking items based on how often items are chosen among all users. **(2) BPR** [16]: a standard bayesian personalized ranking approach. **(3) SBPR** [22]: an extension of BPR with explicit social information.

Evaluation Metrics. A popular metric - Root Mean Square Error (RMSE) is used to evaluate the performances of methods in ratings prediction:

$$RMSE = \sqrt{\frac{\sum_{r_{ij} \in R_{test}} (r_{ij} - \widehat{r}_{ij})^2}{|R_{test}|}}$$

where r_{ij} is the rating user i gave to item j in test set R_{test}, $\widehat{r}_{ij}$ represents the corresponding prediction score by a specific method. A smaller RMSE value means a better performance. Besides, a widely used metric– Area Under the Curve (AUC)–is used to measure the recommendation qualities of methods in items ranking. Let $P_{u,train}$ and $P_{u,test}$ denote the sets of P_u in training and test, respectively. The average AUC is defined as:

$$AUC = \frac{1}{|U|} \sum_{u \in U} \frac{1}{|E(u)|} \sum_{(i,j) \in E(u)} \delta(x_{ui} > x_{uj})$$

where $E(u) \equiv \{(i,j) | i \in (P_{u,test}) \cap (j \notin (P_{u,train} \cup P_{u,test}))\}$. We randomly select a fixed number (1% of the number of all items) of negative item j for evaluation. A larger AUC score means a better result.

4.2 Performance Comparison

The parameters are set either according to the suggestion in previous works or by experimental selection. Specifically, the parameters of SkipGram are set as: L = 20, T = 30, l = 20 and τ = 5. The dimension of latent features d = 5 or 10 for all models. The other settings are: (1) λ (regularization parameter) = 0.01 for NMF, EMF, IMF and CUNE-MF; (2) α (weight of social constraint) = 0.001 for EMF, IMF and 0.01 for CUNE-MF; (3) λ_ω (regularization parameter) = 0.001 for BPR, SBPR and CUNE-BPR; (4) s (social weight) = 2 for SBPR and CUNE-BPR; (5) the number of computed friends k = 50 for CUNE-MF and CUNE-BPR. We randomly select 70% or 50% of the dataset as a training set to train the model, and further predict the remaining 30% or 50% of the dataset. The experimental results are shown in Table 2. We focus on the comparison between our methods and the corresponding models with explicit social information, and report the improvements (%) of (d) over (c) and (4) over (3) in the Table. The main findings are summarized as follows:

- CUNE-MF has the best performance (in bold) in predicting users' ratings. The average RMSE scores of its predictions (d = 10) are around 0.82, 1.14, 1.16 for FilmTrust, CiaoDVD and Epinions datasets, respectively. Besides, CUNE-BPR performs best (in bold) in items ranking. The average AUC values of its results (d = 10) are around 0.98, 0.92, 0.93 for FilmTrust, CiaoDVD and Epinions datasets, respectively.
- CUNE-MF performs better than NMF, demonstrating the effectiveness of extending matrix factorization by semantic social regularization. In addition, CUNE-MF outperforms EMF by an average improvement of 3.0% (d = 10). It validates that the users' top-k semantic friends generated by CUNE are more effective to help shape the latent space of users in MF than the explicit social connections.
- CUNE-BPR significantly outperforms BPR, showing the large benefit of incorporating semantic social feedbacks into BPR. In addition, CUNE-BPR outperforms SBPR by an average improvement of 9.1% on AUC values (d = 10). It verifies that the user's top-k semantic friends generated by CUNE provide more and

Table 2: Performance comparison of our methods and other counterparts. CUNE-MF reaches the best result in ratings prediction and CUNE-BPR performs best in items ranking.

Data	Train	d	Ratings Prediction – RMSE (smaller is better)					Items Ranking – AUC (larger is better)				
			(a) **NMF**	*(b)* **IMF**	*(c)* **EMF**	*(d)* **CUNE-MF**	**improv. d vs. c**	**(1) MP**	**(2) BPR**	**(3) SBPR**	**(4) CUNE-BPR**	**improv. 4 vs. 3**
FilmT.	70%	5	0.878	0.869	0.875	**0.852**	2.63%	0.915	0.936	0.940	**0.968**	2.98%
		10	0.839	0.831	0.836	**0.815**	2.51%	0.915	0.935	0.938	**0.976**	4.05%
	50%	5	0.908	0.890	0.899	**0.873**	2.89%	0.908	0.931	0.935	**0.967**	3.42%
		10	0.855	0.841	0.849	**0.826**	2.71%	0.908	0.930	0.932	**0.974**	4.51%
CiaoD.	70%	5	1.458	1.415	1.449	**1.402**	3.24%	0.761	0.803	0.831	**0.886**	6.62%
		10	1.165	1.147	1.161	**1.125**	3.10%	0.761	0.802	0.846	**0.925**	9.34%
	50%	5	1.535	1.475	1.527	**1.464**	4.13%	0.756	0.798	0.824	**0.873**	5.95%
		10	1.210	1.179	1.203	**1.162**	3.41%	0.756	0.799	0.832	**0.910**	9.38%
Epin.	70%	5	1.363	1.353	1.359	**1.334**	1.84%	0.733	0.792	0.826	**0.908**	9.93%
		10	1.209	1.185	1.202	**1.163**	3.24%	0.733	0.793	0.831	**0.929**	11.8%
	50%	5	1.365	1.356	1.360	**1.335**	1.84%	0.720	0.784	0.792	**0.901**	13.8%
		10	1.212	1.192	1.201	**1.166**	2.91%	0.720	0.783	0.798	**0.924**	15.8%

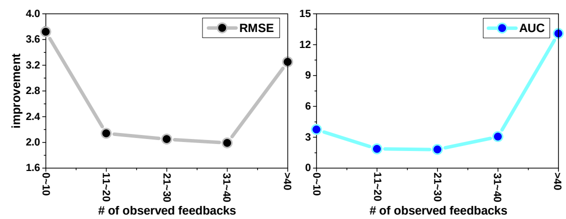

Figure 4: Improvements of CUNE-MF over EMF (% measured in RMSE) and CUNE-BPR over SBPR (% measured in AUC) in different users groups. CUNE-MF and CUNE-BPR reach larger improvements in the first ("cold-start") and the last (active) groups of users than the other groups.

better social feedbacks for improving BPR than the explicit social relationship.

4.3 Analysis and Discussion

Performances on Different Users. As discussed in Section 2, explicit social connections provide limited information on helping the recommendation for "cold-start" users, and introduce potential noises to the recommendation for active users. In order to validate the effectiveness of our methods on improving the recommendation for "cold-start" and active users, we evaluate the improvements of CUNE-MF over EMF (in terms of RMSE) and CUNE-BPR over SBPR (in terms of AUC) in different user groups. We first group all users into 5 classes (i.e. 0∼10, 11∼20, 21∼30, 31∼40 and >40) based on the number of observed feedbacks they have in training sets, and then measure the corresponding improvement in each group. The results of the experiment conducted in the FilmTrust dataset (d = 10, $Train\%$ = 70) are shown in Figure 4. We find that CUNE-MF consistently performs better than EMF and

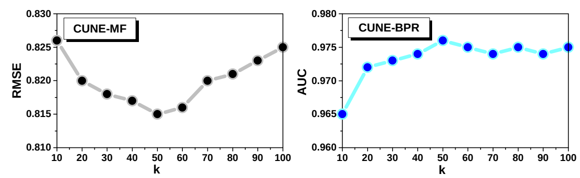

Figure 5: The impact of k on the performances of our models. CUNE-MF or CUNE-BPR reaches the best result when $k = 50$.

CUNE-BPR consistently outperforms SBPR in all user groups. More importantly, our methods achieve relatively larger improvements in the first ("cold-start") and the last (active) user groups as opposed to the other groups. These evaluation results show that, (1) CUNE generates the top-k semantic friends of each user considering both direct and indirect correlations of users, and thus captures the comprehensive correlation between an active user and their neighbors. (2) The top-k semantic friends provide useful information for characterizing the tastes of the "cold-start" users.

Sensitive Analysis of Parameter k. The parameter k plays an important role in CUNE, as it determines how many semantic friends are generated for each user. On the one hand, if we use a large k, the semantic friends in the bottom of list may introduce noise for the training of recommendation models since they are not quite similar to the target user. On the other hand, a small k introduces too few semantic friends who may not provide enough information for shaping the latent feature of target user. We investigate the impact of k on the performances of our models in the FilmTrust dataset with $Train\%$ = 70 and d = 10, and report it in Figure 5. As we can see, with the increment of k, RMSE (of CUNE-MF) decreases at first since more semantic

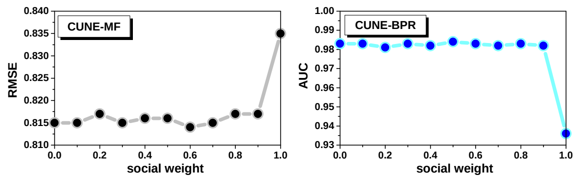

Figure 6: The performances of CUNE-MF (RMSE) and CUNE-BPR (AUC) with different μ. The performances have little change when $\mu < 1$, but significant decreases when $\mu = 1$.

friends are more helpful. But when k goes beyond a certain threshold, RMSE increases with the further increment of k due to the possible involvement of non-similar friends in top-k. Similarly, AUC (of CUNE-BPR) increases at first and decreases slightly when k is larger than the threshold. The existence of the certain point of k ($k = 50$) confirms that an appropriate number of semantic friends results in optimal performance. The sensitive analyses of parameters α and s are illustrated in previous works [11] and [22] respectively. Similarly, the certain value of parameter α in CUNE-MF or parameter s in CUNE-BPR leads to the optimal result.

Explicit Neighbors v.s. Top-k Semantic Friends. CUNE generates the top-k semantic friends of each user only using user-item feedbacks. It is natural to ask whether explicit social connections help improve the performance of CUNE-MF or CUNE-BPR. To answer this question, we introduce explicit social information to compute similarities among users. Formally, let sim_{ij} denote similarity between user i and j w.r.t. feature embeddings, and define a modified similarity $\hat{sim}_{ij} = \mu \cdot t_{i \to j} + (1 - \mu) \cdot sim_{ij}$, where $t_{i \to j}$ denotes the trust value of i on j and parameter μ controls the extent of incorporating explicit social information. Figure 6 reports the performances of CUNE-MF and CUNE-BPR with different settings of μ in the FilmTrust dataset ($d = 10$, $Train\% = 70$), respectively. The figure shows that CUNE-MF or CUNE-BPR has similar performances on different settings of μ when $\mu < 1$, but has worse performance when $\mu = 1$. Note that CUNE-MF or CUNE-BPR degenerates to EMF or SBPR when $\mu = 1$. Therefore, a conclusion can be drawn: explicit social information itself provides little help to our methods for improving the identification of top-k semantic friends. Furthermore, we find that only a small number (less than 2%) of explicit trust links are connections in the top-k semantic friends networks. In other words, users are explicitly connected to only a few of their top-k semantic friends, and the bulk of their trust networks is quite different from their top-k semantic friends networks.

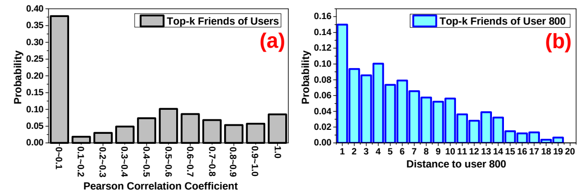

Figure 7: (a) Distribution of PCC between a user and their top-k semantic friends. Around 35% of PCC scores are smaller than 0.1. (b) Distribution of distances between user-800 and their top-k semantic friends in selected walks. Most of the distances are larger than 1.

Case Study of Top-k Semantic Friends. Besides CUNE, IMF (one of the comparison methods) identifies also the top-k friends of users, through computing the Pearson Correlation Coefficient (PCC) scores [9] between a specific user and others based on their ratings, and then selecting the k users who have the largest PCC scores. Due to the requirement of PCC score calculation, such identified top-k friends include only users who have rated items in common, and exclude a large number of users who had no items rated in common. We investigate how our top-k semantic friends cover users with different PCC scores. For each user, we compute the average PCC scores over all their top-k semantic friends. Figure 7 (a) reports the distribution of obtained PCC scores in the FilmTrust dataset ($Train\% = 70$). We see that around 35% of obtained PCC scores are smaller than 0.1. That is to say, if selecting top-k friends by PCC scores, a large portion of semantically relevant users are missed. In addition, Figure 7 (b) shows the distribution of distances between a randomly selected user with ID number 800 and their top-k semantic friends in selected random walks which contain both user-800 and at least one of their top-k friends. There are over 80% of top-k semantic friends whose distances to user-800 are larger than 1, or in other words, who are not directly connected. These two distributions demonstrate that top-k semantic friends generated by CUNE cover both neighboring and non-neighboring users who have direct or transitive correlations.

5 Related Work

MF is popularly used in ratings prediction, as well as the variants of MF [5, 6, 7, 18]. BPR and its variants are proposed for solving items ranking problem [8, 15, 16, 21]. To enhance recommender systems with explicit social information, novel social recommendation models are studied in [3, 4, 9, 10, 11, 14, 20, 22] based on MF or BPR.

Specifically, Ma et al. [10] proposed a social trust ensemble method which linearly combines the basic MF model with a trust-based social neighborhood model. In [3], the authors proposed a trust-based MF technique by considering both the explicit and implicit influences of the neighborhood structure of user trust and the user-item ratings. Yang et al. [20] designed a hybrid MF model that combines both a truster model and a trustee model with the assumption that a specific user's truster and trustee will affect their ratings. In [11], a user social regularization term is introduced to constrain the basic MF. Zhao et al. [22] extended BPR by introducing social positive feedbacks and proposed the SBPR algorithm which achieves better performance in items ranking than BPR.

Our work in this paper differs from the above-mentioned work on extracting implicit and reliable social information from user feedbacks that are easy to access, rather than relying on explicit social information that is difficult to obtain, sparse and noisy to use.

6 Conclusion

In this paper, in order to address the limitations inherent in explicit social information and improve traditional model-based recommendation techniques, we design a novel CUNE method to effectively identify the users' top-k semantic friends only using information from user-item feedbacks. With the top-k semantic friends of each user, we further propose the CUNE-MF and CUNE-BPR algorithms to deal with the problems of ratings prediction and items ranking, respectively. The experimental results on the three real datasets show that our methods outperform the traditional model-based methods as well as the explicit social recommendation models. Overall, our work provides new insight for identifying users' similar friends in recommender systems, and may shed some light on broader research concerning social recommender systems.

Acknowledgement

Research reported in this publication was partially supported by the US Institute of Museum and Library Services (IMLS) National Leadership Grant #LG-81-16-0025, and the King Abdullah University of Science and Technology (KAUST).

References

[1] A. J. Chaney, D. M. Blei, and T. Eliassi-Rad. A probabilistic model for using social networks in personalized item recommendation. In *RecSys*, 2015.

[2] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *KDD*, 2016.

[3] 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.

[4] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*, 2010.

[5] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD*, 2008.

[6] Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

[7] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, 2001.

[8] Y. Liu, P. Zhao, A. Sun, and C. Miao. A boosting algorithm for item recommendation with implicit feedback. In *AAAI*, 2015.

[9] H. Ma. An experimental study on implicit social recommendation. In *SIGIR*, 2013.

[10] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *SIGIR*, 2009.

[11] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *WSDM*, 2011.

[12] T. Mikolov and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[13] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.

[14] D. Rafailidis and F. Crestani. Collaborative ranking with social relationships for top-n recommendations. In *SIGIR*, 2016.

[15] S. Rendle and C. Freudenthaler. Improving pairwise learning for item recommendation from implicit feedback. In *WSDM*, 2014.

[16] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, 2009.

[17] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *ICML*, 2008.

[18] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS*, 2011.

[19] D. Wang, P. Cui, and W. Zhu. Structural deep network embedding. In *KDD*, 2016.

[20] B. Yang, Y. Lei, D. Liu, and J. Liu. Social collaborative filtering by trust. In *IJCAI*, 2013.

[21] L. Yu, G. Zhou, C. Zhang, J. Huang, C. Liu, and Z.-K. Zhang. Rankmbpr: Rank-aware mutual bayesian personalized ranking for item recommendation. In *WAIM*, 2016.

[22] T. Zhao, J. McAuley, and I. King. Leveraging social connections to improve personalized ranking for collaborative filtering. In *CIKM*, 2014.

[23] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang. Bipartite network projection and personal recommendation. *Physical Review E*, 76(4):046115, 2007.