Social Recommendation in Heterogeneous Evolving Relation Network

Bo Jiang [1], Zhigang Lu [1,2], Yuling Liu [1,2], Ning Li[1], Zelin Cui[1]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{jiangbo,luzhigang,liuyuling,liujunrong}@iie.ac.cn

Abstract. The appearance and growth of social networking brings an exponential growth of information. One of the main solutions proposed for this information overload problem are recommender systems, which provide personalized results. Most existing social recommendation approaches consider relation information to improve recommendation performance in the static context. However, relations are likely to evolve over time in the dynamic network. Therefore, temporal information is an essential ingredient to making social recommendation. In this paper, we propose a novel social recommendation model based on evolving relation network, named SoERec. The learned evolving relation network is a heterogeneous information network, where the strength of relation between users is a sum of the influence of all historical events. We incorporate temporally evolving relations into the recommendation algorithm. We empirically evaluate the proposed method on two widely-used datasets. Experimental results show that the proposed model outperforms the state-of-the-art social recommendation methods.

Keywords: Social recommendation · Dynamic evolving · Relation network · Network embedding.

1 Introduction

The last decades have witnessed the booming of social networking such as Twitter and Facebook. User-generated content such as text, images, and videos has been posted by users on these platforms. Social users is suffering from information overload. Fortunately, recommender systems provide a useful tool, which not only help users to select the relevant part of online information, but also discovery user preference and promote popular item, etc. Among existing techniques, collaborative filtering (CF) is a representative model, which attempt to utilize the available user-item rating data to make predictions about the users preferences. These approaches can be divided into two groups [1]: memory-based and model-based. Memory-based approaches [9, 2, 17] make predictions based on the similarities between users or items, while model-based approaches [10, 3] design a prediction model from rating data by using machine learning. Both memory-based and model-based CF approaches have two challenges: data sparsity and

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41

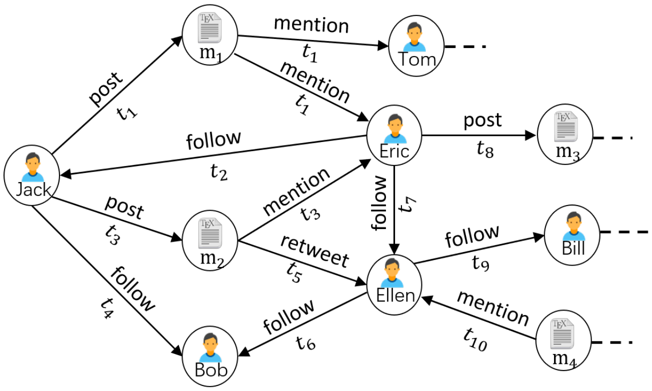


Fig. 1: An illustration of social connections that can change and evolve over time.

cold start, which greatly reduce their performance. In particular, matrix factorization based models [14, 20] have gained popularity in recent years due to their relatively high accuracy and personalized advice.

Existing research works have contributed improvements in social recommendation tasks. However, these approaches only consider static social contextual information. In the real world, knowledge is often time-labeled and will change significantly over time. Figure 1 shows the entire social contextual information over time which can be derived from links on social networks. User *Jack* post message m_1, which mention users *Tom* and *Eric*, at time point t_1. Subsequently, user *Jack* post message m_2, which mention user *Eric* again, at time point t_3. Meanwhile, message m_2 is retweeted by user *Ellen* at time point t_5. We observe that new social action is often influenced by historical related behaviors. In addition, historical behaviors have an impact on current action over time, and the impact strength decreases with time. On the other hand, we notice that the evolving relation network is very sparse, which greatly reduce the recommendation performance. In order to deal with data sparsity, we leverage network embedding technology, which has contributed improvements in many applications, such as link prediction, clustering, and visual.

In this work, we propose a novel social recommendation model based on evolving relation network, named SoERec, which leverages evolving relation network and network embedding technique. The proposed method explicitly models the strength of relations between pair of users learned from an evolving relation network. To efficiently learn heterogeneous relations, network embedding is employed to represent relation into a unified vector space. We conduct experiments on two widely-used datasets and the experimental results show that our proposed model outperforms the state-of-the-art recommendation methods.

The main contributions of this paper are as follows:

- We construct a dynamic, directed and weighted heterogeneous evolving network that contains multiple objects and links types from social network. Compared with static relation graph, the evolving graph can precisely measure the strength of relations.
- We propose a novel social recommendation model by jointly embedding representations of fine-grained relations from historical events based on heterogeneous evolving network.
- We conduct several analysis experiments with two real-world social network datasets, the experimental results demonstrate our proposed model outperforms state-of-the art comparison methods.

The rest of this paper is organized as follows. Section 2 formulates the problem of social recommendation. Section 3 proposes the method of social recommendation based on evolving relation network to recommend the candidate users. Section 4 presents experimental results of recommendation. Finally, Section 5 reviews the related work and Section 6 concludes.

2 Related Work

We briefly review the related works from two lines in this section: one on network embedding and the other on social recommendation.

Network Embedding. Network embedding has been extensively studied to learn a low-dimensional vector representation for each node, and implicitly capture the meaningful topological proximity, and reveal semantic relations among nodes in recent years. The early-stage studies only focus on the embedding representation learning of network structure [30, 25, 15, 22]. Subsequently, network node incorporating the external information like the text content and label information can boost the quality of network embedding representation and improve the learning performance [29, 23, 4, 8, 21, 24]. Network embedding indeed can alleviate the data sparsity and improve the performance of node learning successfully. Therefore, this technique has been effectively applied, such as link prediction, personalized recommendation and community discovery.

Social Recommendation. Recommender systems are used as an efficient tool for dealing with the information overload problem. Various methods of social recommendation have been proposed from different perspectives in recent years. including user-item rating matrix [16], network structure [12], trust relationship [11, 19, 5, 28], individual and friends' preferences [13, 7], social information [26] and combinations of different features [27, 20]. The above social recommendation methods are proposed based on collaborative filtering. These methods all focus on fitting the user-item rating matrix using low-rank approximations, and also use all kinds of social contextual information to make further predictions. Most of the studies that use both ratings and structure deal with static snapshots of networks, and they don't consider the dynamic changes occurring over users' relations. Incorporating temporally evolving relations into the analysis can offer useful insights about the changes in the recommendation task.

DOI: 10.1007/978-3-030-50371-0_41

3 Problem Statement

The intuition behind is that there are two basic accepted observations in a real world: (1) The current behavior of user is influenced by all his/her historical patterns. (2) A behavior with an earlier generation time has a smaller influence on the user's current behavior, while the one with a later generation time has a greater influence. Therefore, we first formally define the concept of *Evolving Relation Network*, as follows:

Definition 1. ***(Heterogeneous Evolving Network)****. A heterogeneous evolving network can be defined as $G = (N_u \cup N_i, E)$, where N_u is the set of vertices representing users, and N_i is the set of vertices representing items, and E is the set of edges between the vertices. The types of edges can be divided into user-user and user-item relationships with temporal information. Hence, G is a dynamic, directed and weighted heterogeneous evolving network.*

From the definition, we can see that each edge not only is an ordered pair from a node to another node, but also has a weight with time-dependent. In order to measure the strength of relations between two nodes objects in the heterogeneous evolving network G, we introduce the concept of *evolving strength*, which is formally defined as follows:

Definition 2. ***(Evolving Strength)****. Given an event $e(\psi, t)$ where ψ is an event type (e.g., post, mention, follow, etc.) and t is the timestamp of e. An event sequence Γ between two nodes is a list of events $\{e_1, e_2, \cdots, e_n\}$, ordered by their timestamps $\{t_1, t_2, \cdots, t_n\}$, where $t_i \leq t_j (1 \leq i < j \leq n)$. An event corresponding to an edge. Thus, the strength of evolving relations denoted by F is the sum of individual event influence.*

We formulate the problem of *social recommendation* as a ranking based task in this work, as follows:

Definition 3. ***(Social Recommendation Problem)****. Given a heterogeneous evolving network G at time t, and a target user u_i, and a candidate set of items Ψ, we aim to generate a top K ranked list of items $\Omega \in \Psi$ for u_i at time $t + 1$ according to the target user's preference inferred from historical feedbacks.*

4 The Proposed Social Recommendation Model

4.1 Probabilistic Matrix Factorization

Let $R \in \mathbb{R}^{M \times N}$ be the rating matrix with M users and N items. The (i, j)-th entry of the matrix is denoted by R_{ij} that represent the rating of user i for item j. $U \in \mathbb{R}^{K \times M}$ and $V \in \mathbb{R}^{K \times N}$ be user and item latent feature matrices respectively, where K is the dimension of latent factors. The preference of i-th user is represented by vector $U_i \in \mathbb{R}^{K \times 1}$ and the characteristic of j-th item is

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41

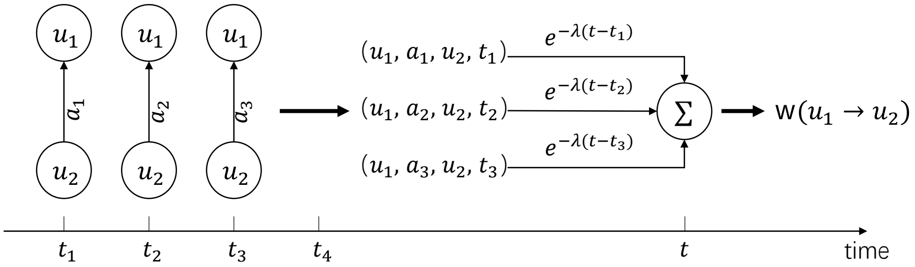


Fig. 2: The evolving strength of relation between pair of users over time.

represented by vector $V_j \in \mathbb{R}^{K\times 1}$. The dot product of U and V can approximate the rating: $\hat{R} \approx U^T V_j$. Recommendation based on Probabilistic Matrix Factorization (PMF) [16] solve the following problem

$$\min_{U,V} \sum_{i=1}^{M} \sum_{j=1}^{N} W_{ij}(R_{ij} - U_i^T V_j)^2 + \gamma(\|U\|_F^2 + \|V\|_F^2) \tag{1}$$

where $W \in \mathbb{R}^{M\times N}$ is a weight matrix. In this work, we set $W_{ij} = 1$ if $R_{ij} \neq 0$ and 0 otherwise. $\gamma > 0$ is the regularization parameter. $(\|U\|_F^2 + \|V\|_F^2)$ can avoid overfitting, $||\cdot||_F$ denotes the Frobenius norm of the matrix.

4.2 Modeling Relation Strength

Incorporating the knowledge from present and historical behavior data can accurately measure the strength of influence, as shown Figure 2. In this work, we model the strength of relation between users as a sum of the influence of each event by multiplying a weight. The weight is calculated by a function, called *decay function*. Since the influence between users can't be less than zero in social networks, the weight ranges from 0 to 1 and decreases with the event's existing time. Thus, we formalize the *decay function* $d_{ij}(t)$ with timestamped information as follows:

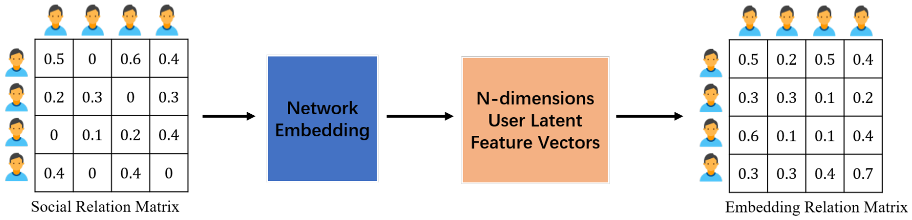
$$d_{ij}(t) = e^{-\lambda(t-t_i)} \tag{2}$$

where t is the current time, t_i is the the generation time of historical event, and λ is a parameter which controls the decay rate. Through the analyses in the following experiments in the paper, we set the parameter λ as 0.6.

Based on the influence of historical events, we can measure the current strength of social relation between users as follows:

$$w_{ij} = \sum_{t_i=0}^{T} d_{ij}(t_i) \cdot I_{e(\psi,t_i)} \tag{3}$$

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41


Network Embedding
N-dimensions User Latent Feature Vectors
Social Relation Matrix
Embedding Relation Matrix

0.5	0	0.6	0.4
0.2	0.3	0	0.3
0	0.1	0.2	0.4
0.4	0	0.4	0

0.5	0.2	0.5	0.4
0.3	0.3	0.1	0.2
0.6	0.1	0.1	0.4
0.3	0.3	0.4	0.7

Fig. 3: The network embedding representation learning of user relation network.

where $I_{e(\psi,t_i)}$ is a parameter which controls the weight of different events. To simplify the model, we assume that the importance of any events is equal.

4.3 Embedding Relation Network

The learned evolving relation network has three characteristics: (1) a weighted and directed graph; (2) a sparsity graph; (3) heterogeneous information network. In order to learn the evolving relation network, we employ large-scale information network embedding (LINE) [18] model to simultaneously retain the local and global structures of the network. In particular, we leverage the LINE model to learn users' embedded representations of the evolving relation network the first-order proximity and the second-order proximity. As shown Figure 3, the detailed process is demonstrated as follows.

User Relation with First-order Proximity. The first-order similarity can represent the relation by the directly connected edge between vertices. We model the joint probability distribution of users u_i and u_j as the first-order similarity $p_1(u_i, u_j)$. The similarity can be defined as follows:

$$p_1(u_i, u_j) = \frac{1}{1 + exp(-\vec{u}_i^T \vec{u}_j)} \tag{4}$$

where $\vec{u}_i \in \mathbb{R}^d$ is the low-dimensional vector representations of vertices u_i. The empirical distribution between vertices u_i and u_j is defined as follows:

$$\hat{p}_1(u_i, u_j) = \frac{w_{ij}}{W} \tag{5}$$

where $W = \sum_{(u_i,u_j)\in E} w_{ij}$, and w_{ij} is the relation strength of the edge (u_i, u_j) measured by Equation (3). To preserve the first-order proximity in evolving relation network, we use the KL-divergence to minimize the joint probability distribution and the empirical probability distribution as follows:

$$O_1 = -\sum_{(u_i,u_j)\in E} w_{ij} log p_1(u_i, u_j) \tag{6}$$

User Relation with Second-order Proximity. The second-order proximity assumes that vertices sharing many connections to other vertices are similar to each other. In this work, we assume that two users with similar neighbors have high similarity scores between them. Specifically, we consider each user vertex as a specific "context", and users with similar distributions over the "contexts" are assumed to be similar. Thus, each user vertex respectively plays two roles: the user vertex itself and the specific "context" of other user vertices. We introduce two vectors $\vec{u}_i$ and $\vec{u}_i^{'}$, where $\vec{u}_i$ is the representation of u_i when it is treated as a vertex, and $\vec{u}_i^{'}$ is the representation of u_i when it is treated as a specific "context". For each directed user edge (u_i, u_j), we firstly define the probability distribution of "context" u_j generated by user vertex u_i as follows:

$$p_2(u_j|u_i) = \frac{exp(\vec{u}_j^{'T}\vec{u}_i)}{\sum_{k=1}^{\mathcal{K}} exp(\vec{u}_k^{'T}\vec{u}_i)} \tag{7}$$

where $\mathcal{K}$ is the number of user vertices or "contexts". The empirical distribution of "contexts" u_j generated by user vertex u_i is defined as:

$$\hat{p}_2(u_j|u_i) = \frac{w_{ij}}{d_i} \tag{8}$$

where w_{ij} is the weight of the edge (u_i, u_j) as the same, and d_i is the out-degree of vertex u_i, i.e. $d_i = \sum_{k \in N(i)} w_{ik}$, with $N(i)$ as the set of out-neighbors of u_i.

To preserve the second-order user relation, the following objective function is obtained by utilizing the KL-divergence:

$$O_2 = -\sum_{(u_i,u_j)\in E} w_{ij} log p_2(u_j|u_i) \tag{9}$$

Combining First-order and Second-order Proximities. To embed the evolving network by preserving both the first-order and second-order proximities, LINE model can minimize the objective functions O_1 and O_2 respectively, and learns two low-dimensional representations for each user vertex. Then, the two low-dimensional representations are concatenated as one low-dimensional feature vector to simultaneously preserve the local and global structures of evolving relation network. Finally, each user vertex u_i is represented as $\vec{U} \in \mathbb{R}^{d_1+d_2}$.

4.4 Evolving Relation Embedding Recommendation Model

Incorporating simultaneously user's explicit relation and implicit relation can boost the ability of social recommendation. As mentioned above, LINE model can learn users' embedded representations, where first-order proximity correspond to the strength of explicit relation and second-order proximity correspond to the strength of implicit relation. Hence, the fine-grained relation measure can better predict user ratings by also encoding both the first-order and second-order relationships among users.

After performing the LINE model, we can obtain users' embedded presentations. We then measure the fine-grained relations among users on the basis of

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41

the inner product of the presentations as follows:

$$s_{ij} = \frac{\vec{u}_i^T \vec{u}_j}{\|\vec{u}_i\| \|\vec{u}_j\|} \tag{10}$$

where $\vec{u}_i$ and $\vec{u}_j$ denote the low-dimensional feature representations of users u_i and u_j, respectively. In this work, relation strength w_{ij} can be viewed as a coarse-grained relation value between users u_i and u_j. Compared to coarse-grained measure, the fine-grained measure s_{ij} is more informative, and can effectively distinguish the importance of recent and old events among users. In other words, the fine-grained measure can deduce the strength of latent relation based on neighborhood structures while two users have no explicit connections.

The fact of matter is that user decision making is influenced by his/her own preferences and close friends in real-world situations. Specifically, on the one hand, users often have different preferences for different items. On the other hand, user are likely to accept their friends' recommendations. Thus, we assume that the final rating of user u_i for item v_j is a linear combination between the user's own preference and his/her friends' preferences, where the rating can be defined as follows:

$$\hat{R}_{ij} = \eta U_i^T V_j + (1-\eta) \sum_{k \in S(u_i)} s_{ik} U_k^T V_j \tag{11}$$

where $S(u_i)$ is the set of most intimate friends of user u_i. In the above equation, the first item corresponds to the prediction rating based on their own preferences, while the second item corresponds to the prediction rating based on the preferences of his/her friends, and η is a parameter that controls the relative weight between user's own preferences and friends' preferences.

The ratings of users to items are generally represented by an ordered set, such as discrete values or continuous numbers within a certain range. In this work, without loss of generality, the differences in the users' individual rating scales can be considered by normalizing ratings with a function $f(x)$:

$$f(x) = \frac{x - R_{min}}{R_{max} - R_{min}} \tag{12}$$

where R_{max} and R_{min} represent the maximum and minimum ratings, respectively. $f(x)$ values can be fell in the [0,1] interval. Meanwhile, we use the logistic function $g(x) = 1/(1+e^{-x})$ to limit the predicted ratings $\hat{R}_{ij}$ within the range of [0,1].

Based on this, the task of social recommendation is likewise to minimize the predictive error. Hence, the objective function of the evolving relation embedding recommendation algorithm is formalized as:

$$\begin{aligned} \mathcal{L} = \frac{1}{2} \min_{U,V} \sum_{i=1}^{M} \sum_{j=1}^{N} W_{ij} (R_{ij} - g(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_k^T V_j))^2 \\ + \frac{\gamma}{2} (\|U\|_F^2 + \|V\|_F^2) \end{aligned} \tag{13}$$

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41

where $S(u_i) = \{k|s_{ik} \geq \epsilon\}$ is the set of most intimate friends of user u_i, and the parameter ϵ is the threshold of the close relation value.

We adopt stochastic gradient descent (SGD) to solve the local minimum solution of $\mathcal{L}$, and learn the latent feature vectors U_i and V_j. The partial derivatives of the objective function $\mathcal{L}$ with respect to U_i and V_j are computed as:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial U_i} = {} & \alpha \sum_{j=1}^{N} W_{ij} g'(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_i^T V_j) V_j \\ & \times (g(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_i^T V_j) - R_{ij}) \\ & + (1-\alpha) \sum_{j=1}^{N} \sum_{p \in S(u_i)} W_{ij} g'(\alpha U_p^T V_j + (1-\alpha) \sum_{q \in S(p)} s_{pq} U_q^T V_j) \\ & \times (g(\alpha U_p^T V_j + (1-\alpha) \sum_{q \in S(p)} s_{pq} U_q^T V_j) - R_{pi}) s_{pi} V_j + \gamma U_i \end{aligned} \tag{14}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial V_j} = {} & \alpha \sum_{i=1}^{M} W_{ij} g'(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_i^T V_j) U_i \\ & \times (g(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_i^T V_j) - R_i) \\ & \times (\alpha U_i + (1-\alpha) \sum_{k \in S(u_i)} s_{ik} U_k) + \gamma V_j \end{aligned} \tag{15}$$

where $g'(x) = e^{-x}/(1+e^{-x})^2$ is the derivative of the logistic function $g(x)$.

5 Experiments

In this section, we first describe experimental datasets and metrics. We then present the baselines and the experiments settings. Finally, we give the experimental results and analyze them.

5.1 Datasets

To evaluate the proposed model, we use two real-world datasets for this task: Weibo and Last.fm.

Weibo Dataset[1]. The data is collected from Sina Weibo, which is the most popular microblogging platform in China. It includes basic information about messages (time, user ID, message ID etc.), mentions (user IDs appearing in messages), forwarding paths, and whether containing embedded URLs or event keywords. In addition, it also contains a snapshot of the following network of users (based on user IDs).

[1] https://www.aminer.cn/influencelocality

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50371-0_41

Table 1: Statistics of the datasets.

Dataset	Weibo	Last.fm
#User	840,432	1,892
#Item	30,000	17,632
#User-User Relations	154,352,856	12,717
#User-Item Relations	355,754	92,834
Density	0.014%	0.71%

Last.fm Dataset[2]. This dataset has been obtained from Last.fm online music system. Its users are interconnected in a social network generated from Last.fm "friend" relations. Each user has a list of most listened music artists, tag assignments, i.e. tuples [user, tag, artist], and friend relations within the dataset social network. Each artist has a Last.fm URL and a picture URL.

For two datasets, the user-user relations are constructed from following or bi-directional friendships between social network users, user-item relations are constructed from the user posting or listening behavior. The statistics of two datasets are summerized in Table 1.

5.2 Evaluation Metrics

We use the mean absolute error (MAE), root mean square error (RMSE) and the average precision of top-K recommendation (Average P@K) to evaluate the performance of recommendation algorithms. According to their definition, a smaller MAE/RMSE or bigger Average P@K value means better performance. For each dataset, {40%, 80%} are selected randomly as training set and the rest as the test set. We will repeat the experiments 5 times and report the average performance.

5.3 Comparison Algorithms

In order to evaluate the effectiveness of our proposed recommendation algorithm, we select following recommendation algorithms as comparison methods:

- PMF [16]: The method adopts a probabilistic linear model with Gaussian distribution, and the recommendations are obtained only by relying on the rating matrix of users to items.
- SoRec [12]: The method integrates social network structure and the user-item rating matrix based on probabilistic matrix factorization. However, the algorithm ignore the temporal changes of relations between users.
- RSTE [11]: The model fuses the users' tastes and their trusted friends' favors together for the final predicted ratings. Similarly, the method doesn't consider the changes of trust relations over time.

[2] http://ir.ii.uam.es/hetrec2011/datasets.html

DOI: 10.1007/978-3-030-50371-0_41

Table 2: Performance comparisons of different recommender models.

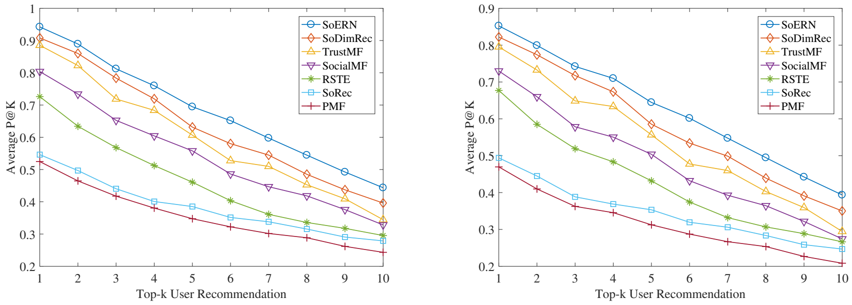
Dataset	**Method**	**MAE(40%)**	**MAE(80%)**	**RMSE(40%)**	**RMSE(80%)**
Weibo	PMF	0.9963	0.9110	1.0346	0.9474
	SoRec	0.9602	0.8957	1.0158	0.9329
	RSTE	0.9319	0.8515	1.0023	0.9301
	SocialMF	0.9044	0.8232	0.9778	0.9168
	TrustMF	0.8879	0.8031	0.9465	0.8885
	SoDimRec	0.8528	0.7884	0.9304	0.8757
	SoERec	**0.8249**	**0.7495**	**0.9128**	**0.8655**
Last.fm	PMF	1.0582	1.0292	1.2691	1.1306
	SoRec	1.0442	1.0996	1.2009	1.0971
	RSTE	1.0386	0.9936	1.1716	1.0876
	SocialMF	1.0299	0.9869	1.1546	1.0801
	TrustMF	1.0076	0.9804	1.1408	1.0718
	SoDimRec	0.9967	0.9768	1.1211	1.0639
	SoERec	**0.9851**	**0.9617**	**1.1092**	**1.0590**

- SocialMF [6]: The model integrates a trust propagation mechanism into PMF to improve the recommendation accuracy. However, the algorithm represents the feature vector of each user only by the feature vectors of his direct neighbors in the social network.
- TrustMF [28]: The model proposes social collaborative filtering recommendations by integrating sparse rating data and social trust network. The algorithm can map users into low-dimensional truster feature space and trustee feature space, respectively.
- SoDimRec [20]: The model adopts simultaneously the heterogeneity of social relations and weak dependency connections in the social network, and employs social dimensions to model social recommendation.

The optimal experimental settings for each method were either determined by our experiments or were taken from the suggestions by previous works. The setting that were taken from previous works include: the learning rate $\eta = 0.001$; and the dimension of the latent vectors $d = 100$. All the regularization parameters for the latent vectors were set to be the same at 0.001.

5.4 Experimental Results

Comparisons of Recommendation Model We use different amounts of training data (40%, 80%) to test the algorithms. Comparison results are demonstrated in Table 2, and we make the following observations: (1) Our proposed approach SoERec always outperforms baseline methods on both MAE and RMSE. The major reason is that the proposed framework exploits heterogeneity of social relations via time dimension and network embedding technique. (2) Recommendation systems by exploiting social relations all perform better than the PMF method only by using user-item rating matrix in terms of both MAE and RMSE.



(a) Weibo Dataset.

(b) Last.fm Dataset.

Fig. 4: The overall average P@K score of each method with different K.

(3) Among these relation-aware recommendation methods, leveraging more indirect relations method generally achieves better performance than only using direct connections methods. In a word, social relations play an important role in context-aware recommendations.

Top-K User Recommendation Figure 4 summarizes the user recommendation performance for the state-of-the-art methods and the proposed model. Generally speaking, it can be shown from the figure that the average P@K value decreases gradually along with the increasing number of K. Besides, we can also observe on both datasets that: Firstly, the proposed method consistently perform better than baseline methods, indicating that the considering cross-time evolving graph embedding by SoERec model can be recommended the more appropriate users than recommendation models without considering time dimension. Secondly, trust-based algorithms (TrustMF, SocialMF and RSTE) consistently perform better than non-trust based benchmarks (SocRec, PMF). It is because trust-based algorithms can fully exploit the network structure, which tackles the incomplete, sparse and noisy problem. Finally, among the different recommendation methods, considering heterogeneous network (SocDimRec and SoERec) significantly performs better than the other methods.

6 CONCLUSION

In this paper, we propose a novel social recommendation model by incorporating cross-time heterogeneity network of relations. We construct an evolving heterogeneous relation network with timestamp information based on multiple objects and links types. The evolving graph can learn more accurate user relations. We then use network embedding technique to encode the latent feature spaces of relations into the objective function. To demonstrate the effective of the proposed model, we construct extensive experiments. The experimental results reveal that our proposed method outperforms the state-of-the-art baseline methods.

Acknowledgement

This work is supported by Natural Science Foundation of China (No.61702508, No.61802404), and National Social Science Foundation of China (No.19BSH022), and National Key Research and Development Program of China (No.2019QY1303). This work is also supported by the Program of Key Laboratory of Network Assessment Technology, the Chinese Academy of Sciences; Program of Beijing Key Laboratory of Network Security and Protection Technology.

References

1. Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge & Data Engineering*, (6):734–749, 2005.
2. Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.
3. Prem Gopalan, Jake M Hofman, and David M Blei. Scalable recommendation with hierarchical poisson factorization. In *UAI*, pages 326–335, 2015.
4. Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2016.
5. Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*, pages 135–142, 2010.
6. Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender Systems*, pages 135–142. ACM, 2010.
7. Meng Jiang, Peng Cui, Rui Liu, Qiang Yang, Fei Wang, Wenwu Zhu, and Shiqiang Yang. Social contextual recommendation. In *CIKM*, pages 45–54, 2012.
8. Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
9. Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, (1):76–80, 2003.
10. Nathan N Liu and Qiang Yang. Eigenrank: a ranking-oriented approach to collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 83–90. ACM, 2008.
11. Hao Ma, Irwin King, and Michael R Lyu. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 203–210. ACM, 2009.
12. Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 931–940. ACM, 2008.
13. Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender systems with social regularization. In *WSDM*, pages 287–296. ACM, 2011.
14. Andriy Mnih and Ruslan R Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2008.

15. Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1105–1114. ACM, 2016.
16. Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1257–1264, 2007.
17. Badrul Munir Sarwar, George Karypis, Joseph A Konstan, John Riedl, et al. Item-based collaborative filtering recommendation algorithms. *WWW*, 1:285–295, 2001.
18. Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1067–1077. International World Wide Web Conferences Steering Committee, 2015.
19. Jiliang Tang, Huiji Gao, and Huan Liu. mtrust:discerning multi-faceted trust in a connected world. pages 93–102, 2012.
20. Jiliang Tang, Suhang Wang, Xia Hu, Dawei Yin, Yingzhou Bi, Yi Chang, and Huan Liu. Recommendation with social dimensions. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
21. Cunchao Tu, Han Liu, Zhiyuan Liu, and Maosong Sun. Cane: Context-aware network embedding for relation modeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1722–1731, 2017.
22. Cunchao Tu, Hao Wang, Xiangkai Zeng, Zhiyuan Liu, and Maosong Sun. Community-enhanced network representation learning for network analysis. *arXiv preprint arXiv:1611.06645*, 2016.
23. Cunchao Tu, Weicheng Zhang, Zhiyuan Liu, Maosong Sun, et al. Max-margin deepwalk: Discriminative learning of network representation. In *IJCAI*, pages 3889–3895, 2016.
24. Cunchao Tu, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. Transnet: translation-based network representation learning for social relation extraction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.
25. Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1225–1234. ACM, 2016.
26. Menghan Wang, Xiaolin Zheng, Yang Yang, and Kun Zhang. Collaborative filtering with social exposure: A modular approach to social recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
27. Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. Social recommendation with strong and weak ties. In *CIKM*, pages 5–14. ACM, 2016.
28. Bo Yang, Yu Lei, Jiming Liu, and Wenjie Li. Social collaborative filtering by trust. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(8):1633–1647, 2016.
29. Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y Chang. Network representation learning with rich text information. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2111–2117, 2015.
30. Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*, pages 587–596. ACM, 2013.