Improving Social Recommendations with Item Relationships

Haifeng Liu[1], Hongfei Lin[1(✉)], Bo Xu[1], Liang Yang[1], Yuan Lin[1(✉)], Yonghe Chu[1], Wenqi Fan[2], and Nan Zhao[1]

[1] Dalian University of Technology, Dalian, Liaoning, China
{liuhaifeng,yhchu,nanzhao}@mail.dlut.edu.cn,
{hflin,xubo,liang,zhlin}@dlut.edu.cn
[2] City University of Hong Kong, Kowloon, Hong Kong, China
wenqifan03@gmail.com

Abstract. Social recommendations have witnessed rapid developments for improving the performance of recommender systems, due to the growing influence of social networks. However, existing social recommendations often ignore to facilitate the substitutable and complementary items to understand items and enhance the recommender systems. We propose a novel graph neural network framework to model the multi-graph data (user-item graph, user-user graph, item-item graph) in social recommendations. In particular, we introduce a viewpoint mechanism to model the relationship between users and items. We conduct an extensive experiment on two public benchmarks, demonstrating significant improvement over several state-of-the-art models.

1 Introduction

Among the existing recommender systems approaches, Collaborative Filtering (CF) is one of the most popular and successful techniques for building advanced recommender systems. The basic idea of CF techniques is that users' preferences can be learned through their history behaviors towards items, such as ratings, clicks, and searching. The majority of CF-based recommender systems can achieve a great performance when huge amounts of history feedback between users and items are available. However, the prediction performance may drop significantly when there exists little interaction between users and items [3], given the fact that recommender systems cannot fully understand users and items through the limited interactions.

Recent studies have shown that social relations among users can provide another stream of potential information for understanding users' preferences, such that the sparse issue can be alleviated and the performance of recommender systems can be further improved. Those methods are known as social recommendations, which are proposed to integrate the information of social relations for enhancing the performance of recommender systems [2]. The rationale behind social recommendations is that users' preferences are similar or influenced by

their social neighbors [3], such as friends and classmates, as suggested by social correlation theories [7]. These social relations have been proven to improve the prediction performance in recommender systems.

Although the aforementioned methods advance social recommendation with GNNs techniques, existing social recommendations methods are still far from satisfactory. Most of them only involve user-item interactions and user-user connections [2,8], while correlations among items could also be of great help. The reason behind this is because items are rather independent and some of them could be similar or related [8].

Moreover, item i_1 and item i_3 are likely to be related since they are co-purchased by user u_1, and item i_1 and i_2 are also likely to be connected or similar as they are bought by two connected social users u_1 and u_2. These relations among items can be formed as the third graph - item-item graph, which can be generated based on users' commonly purchased behaviors or social relations. Thus, it is quite desirable to consider the relations among items for learning better representations of items.

In this paper, we propose a graph neural network framework for the social recommendation (VMRec) by taking advantage of multi-graph and review information. Our major contribution can be summarized as follows,

– We introduce a principle way to utilize the review information to measure the strength among node (users/items) multi-graph (e.g., social graph, user-item graph, and item-item graph) for the social recommendation;
– We propose a GNN-based framework for a social recommendation (VMRec), which integrate review information and multi-graph to learn better user and item representations;

2   The Proposed Framework

In social recommendation systems, there are two set of entities: a user set $U(|U| = M)$ and an item set $I(|I| = N)$, where $M(N)$ is the number of users (items). We define the user-item interactions matrix $\boldsymbol{R} \in \mathbb{R}^{N \times M}$ from user's implicit feedback, where the i, j-th element $r_{i,j}$ is 1 if there is an interaction between user u_i and item v_j, and 0 otherwise. The social graph between users can be described by $\boldsymbol{S} \in \mathbb{S}^{M \times M}$, where $s_{i,j} = 1$ if there is a social relation between user u_i and user u_j, and 0 otherwise. The item-item graph can be denoted as $\boldsymbol{I} \in \mathbb{I}^{N \times N}$, where $I_{i,j} = 1$ if there is a connection between item i_1 and item i_2, and 0 otherwise. Given an interactions matrix $\boldsymbol{R}$, social graph $\boldsymbol{S}$ and item-item graph $\boldsymbol{I}$, we aim to predict the unobserved entries (i.e., where $r_{i,j} = 0$) in $\boldsymbol{R}$.

2.1   Model Architecture

There are three main components in our model, including embedding layer, graph modeling, and prediction modeling. The first component is embedding

layer, which is proposed to combine review text and free embedding for users and items. The second component is graph modeling, which aims to learn the latent factors of users and items from different perspectives. The third component is prediction modeling for finalizing the user and item representation for prediction.

2.2 Embedding Layer

Users can provide their reviews towards items and these reviews can help characterize representations of users and items. Inspired by the Natural Language Processing, we employ the Word2Vec to initial the words in the review sentences, and then take the average of all words representation of reviews for users or items. They can be denoted as $\boldsymbol{p}^r \in \mathbb{R}^{M \times D}$ for users text representation and $\boldsymbol{q}^r \in \mathbb{R}^{N \times D}$ for items text representation with the dimension size D. In addition, we also introduce the free user and item representation, denoted as $\boldsymbol{p}^g \in \mathbb{R}^{M \times D}$ for users and $\boldsymbol{q}^g \in \mathbb{R}^{N \times D}$ for items.

Then, we introduce to fuse these two kinds of representation via neural networks to obtain user representation $\boldsymbol{p}_u^f$ and item representation $\boldsymbol{q}_i^f$ as follows,

$$\boldsymbol{p}_u^f = g(\boldsymbol{W} \times [\boldsymbol{p}_u^r, \boldsymbol{p}_u^g] + \boldsymbol{b}) \tag{1}$$

$$\boldsymbol{q}_i^f = g(\boldsymbol{W} \times [\boldsymbol{q}_i^r, \boldsymbol{q}_i^g] + \boldsymbol{b}) \tag{2}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are a transformation matrix and bias, and $g(*)$ is a non-linear function like ReLU. $[\boldsymbol{p}_u^r, \boldsymbol{p}_u^g]$ is the concatenation operation.

2.3 Graph Modeling

In addition to the user-item graph, the social graph and item-item graph provides a great opportunity to learn user and item representations from different perspectives. The GNNs-based social recommender systems employ graph neural networks to aggregate graph feature of neighboring nodes, which makes the aggravated representation more powerful. However, GNNs-based social recommender systems [3] only design for a single graph or two graphs, real-world scenarios often contain multiple interactive graphs. In this paper, we extend the graph neural network into a multi-graph with viewpoint mechanism for social recommender system.

User-Item Aggregation. To learn representations of users in the user-item graph, since different items contribute differently to user purchased behaviors, we introduce to incorporate the review information to differentiate the importance of items as follows,

$$\boldsymbol{p}_u^p = Aggre_{user-item}\{\sum_{i \in R_u} v_{u,i} \boldsymbol{q}_i^f\} \tag{3}$$

where $v_{u,i}$ is the important weight between user u and item i. In particular, $v_{u,i}$ can be calculated through user and item review information with viewpoint mechanism as follows,

$$v_{u,i} = \exp(-\gamma \|\boldsymbol{p}_u^r - \boldsymbol{q}_i^r\|^2) \tag{4}$$

where $\boldsymbol{p}_u^r$ and $\boldsymbol{q}_i^r$ are the user's review vector and item's review vector. γ is hyper-parameter.

Social Aggregation. The item-item aggregation is similar to social aggregation, here we ignored the item-item part for simply. As suggested by social theories [7], users are more likely to be influenced by their friends [2]. It is important to integrate social relations information into learning user representations. Moreover, tie strengths among users can differently influence users' behaviors. In particular, users in strong tie might share more similar preferences than users in weak tie. To consider these heterogeneous strengths of social relations, we introduce to differentiate users' local neighbors during aggregation operation in graph neural networks as follows,

$$\boldsymbol{p}_u^g = \sigma(\boldsymbol{W} \cdot Aggre_{social}\{\sum_{a \in S_u} v_{u,a}(\boldsymbol{p}_u^g \odot \boldsymbol{p}_a^g)\} + \boldsymbol{b}) \tag{5}$$

where S_u means social friends set of user u, and the $\odot$ denote the element-wise product. Similarly, $v_{u,a}$ is the important weight between user u and his friend a, and $v_{u,a}$ can be calculated through user and social friend review information with viewpoint mechanism as follows,

$$v_{u,a} = \exp(-\gamma \|\boldsymbol{p}_u^r - \boldsymbol{p}_a^r\|^2) \tag{6}$$

where $\boldsymbol{p}_u^r$ and $\boldsymbol{p}_a^r$ are the user's review vector and friend's review vector.

In addition to the directly connected neighbors, distant neighbors can also be beneficial, that is due to the fact that information can be diffused throughout the social network, and users might be affected by the k-hop neighbors. Therefore, we introduce to aggregate social information through k-layer aggregation as follows,

$$\boldsymbol{p}_u^{g^{k+1}} = \sigma(\boldsymbol{W} \cdot Aggre_{social}\{\sum_{a \in S_u} v_{u,a}(\boldsymbol{p}_u^{g^k} \odot \boldsymbol{p}_a^{g^k})\} + \boldsymbol{b}) \tag{7}$$

where $\boldsymbol{p}_u^{g^k}$ denotes the user u representation after k-layer aggregation operation. The user fusion representation $\boldsymbol{p}_u^f$ is equal to $\boldsymbol{p}_u^{g^k}$ when $k=0$.

2.4 Prediction Modeling

The last component in our proposed model is prediction modeling, which aims to finalize the user and item representation for prediction.

As the social graph and user-item graph provide important signals to understand users preferences, we propose to obtain the final user representation $\boldsymbol{p}_u$ as follows:

$$\boldsymbol{p}_u = \boldsymbol{p}_u^f \oplus \boldsymbol{p}_u^{g^{k+1}} \oplus \boldsymbol{p}_u^p \tag{8}$$

$$\boldsymbol{q}_i = \boldsymbol{q}_i^f \oplus \boldsymbol{q}_i^{g^{k+1}} \tag{9}$$

$$r_{ui} = \boldsymbol{p}_u^T \boldsymbol{q}_i \tag{10}$$

where $\oplus$ indicates summation operation. Likewise, we introduce to extract useful information from the item-item graph to enrich the representation of item as Eq. 9. With the user and item representation (e.g., $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$), we perform score prediction via the inner product as Eq. 10.

2.5 Model Training

In order to learn the model parameters of our proposed model, we adopt the pair-wise loss as our objective for the Top-k recommendation task [8] as follows,

$$\min_{\theta} L = \sum_{u=1}^{M} \sum_{(i,j) \in R_u} -\sigma(r_{ui} - r_{uj}) + \lambda \|\theta\|^2 \qquad (11)$$

where $\sigma(\cdot)$ is a sigmoid function. M denote the number of user-item pairs for training. θ denotes all trainable model parameters in our VMRec framework. λ is a regularization parameter that controls the complexity of user and item graph representation matrices. R_u denotes user u interactive items set. By optimizing the loss function, all parameters can be tuned via backward propagation.

3 Experiment Results and Discuss

3.1 Experimental Settings

Datasets. In our experiments, we choose two representative datasets Yelp and Flickr. In these two datasets, the user-item interactions can be seem as the user-item graph, and the user-user connections are regarded as a social graph. Table 1 summary the statistics the two datasets. Similar to many studies [8], we randomly select 90% of data for training, the rest of the data for testing. To tune the parameters, we randomly selected 10% of train data as the validation set.

Table 1. Statistics of the two datasets

Datasets	Yelp	Flickr
# Users	17,237	8,358
# Items	38,342	82,120
# Rating Density	0.031%	0.046%
# User-User Density	0.048%	0.268%
# Item-Item Density	0.001%	0.002%

Baselines. To evaluate the performance, we compared our proposed model VMRec with seven representative baselines, including traditional recommender system without social network information (BPR-MF [6] and SVD++ [4]), tradition social recommender systems (ContextMF [5]), deep neural networks based social recommender systems (TrustSVD [4]), and graph neural network based recommender system (DiffNet [8], GC-MC [1], PinSage [9]). Some of the original baseline implementations (BPR-MF and TrustSVD) are for rating prediction on recommendations. Therefore we adjust their objectives to pair-wise prediction with BPR loss using negative sampling.

Evaluation Metrics. We use two widely-used metrics [8] to evaluate the performance of recommender systems: NDCG@N and HR@N, where N is the number of recommended items. HR@N refers to the ratio of recovered items to the top-N recommended and NDCG@N measures the ranking performance of the recommender list in the testing data.

Table 2. Overall results of different algorithms.

Models	Yelp				Flickr			
	HR		NDCG		HR		NDCG	
	N = 5	N = 10	N = 5	N = 10	N = 5	N = 10	N = 5	N = 10
TrustSVD	0.1906	0.2915	0.1385	0.1738	0.1072	0.1427	0.0970	0.1085
ContextMF	0.2045	0.3043	0.1484	0.1818	0.1095	0.1433	0.0920	0.1102
GC-MC	0.1932	0.2937	0.1420	0.1740	0.0897	0.1182	0.0795	0.0956
PinSage	0.2099	0.3065	0.1536	0.1868	0.0925	0.1242	0.0842	0.0991
DiffNet	0.2276	0.3477	0.1679	0.2121	0.1210	0.1641	0.1142	0.1273
VMRec	**0.2445**	**0.3630**	**0.1772**	**0.2203**	**0.1410**	**0.1688**	**0.1252**	**0.1300**

Parameter Setting. To fairly compare the performance of models, we trained all of the models with the BPR loss for ranking recommendation. We randomly initialized user and item free representation with a Gaussian distribution, where the mean and standard deviation is 0 and 0.01. We used Adam optimizer to optimize all parameters with 0.001 learning rate. The latent factor dimension is set to 64, and $\gamma = \frac{1}{64}$. We used $Relu$ as the activation function in neural network. We set the depth parameter $K = 2$ for the two datasets.

3.2 Performance Comparison

The performance of different recommender system HR@N and NDCG@N is shown in Table 2. We can observe that TrustSVD, and ContextMF perform the worst performance among all the baselines across both datasets, that is due to

the face that they model the user-item interactions via the linear inner product (MF-based methods). Meanwhile, GC-MC and PinSage, which is based on graph neural network architecture, can obtain much better performance than MF-based. Among baselines, DiffNet shows quite strong performance. Note that Diffnet is proposed to harness the power graph neural networks to learn representations in social recommendations. This performance gain implies once more the power of graph neural networks in recommender systems. Our proposed method VMRec consistently outperforms all the baseline methods. In particular, VMRec improves over DiffNet about 10% of HR on both datasets. Compared with Diffnet, we explicitly incorporate the relationships between items for learning items representation.

3.3   Detailed Model Analysis

The Effect of Viewpoint Mechanism. In this subsection, we analyze the effect of Viewpoint Mechanism on the performance of our model by comparing with two variants (VMRec-GCN and VMRec-One). The $v_{u,i}$ in VMRec-GCN is set to $\frac{1}{|R_u|}$ in Eq. (6), while $v_{u,i}$ in VMRec-One is set to 1 for all user-item pair. In VMRec-Viewpoint, we adopt viewpoint mechanism to calculate the $v_{u,i}$ from user and item reviews.
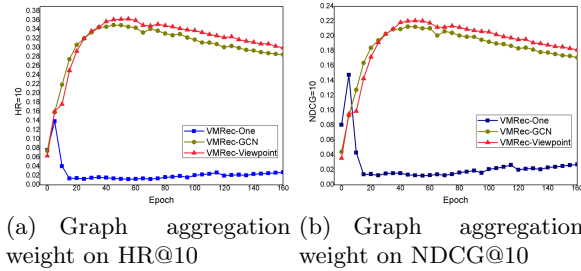

(a) Graph aggregation   (b) Graph aggregation
weight on HR@10   weight on NDCG@10

Fig. 1. Different graph representation process performance on Yelp

Figure 1 shows the performance of different graph aggregation weight on Yelp dataset. We can observe that VMRec-Viewpoint performs better that VMRec-GCN, which implies that user's review information is beneficial to enhance the performance of recommender system. In addition, We can see that VMRec-One without weight information, the performance of item prediction is deteriorated significantly, It justifies our assumption that users' review text can help to learn user or item latent factors and improve the performance of recommendation.

4   Related Work

Among recommender systems, collaborative filtering (CF) methods are the most popular techniques to model the user-item interactions. However, recent studies

have shown that these models are limited to modeling the relationship between users and products in a linear space, and cannot capture the complex, non-linear relationship between users and items. To partially alleviate this issue, NeuMF is proposed to model the complex interactions between user and item representations via deep neural network. The GNNs techniques also applied to jointly user-item interactions and user-user connections in social recommendations [2,8]. Diffnet considers the social recommender systems from information propagation, and employs deep influence propagation model to aggregate users' influence in social graph [8].

5   Conclusion

In this paper, we integrated the relationships among items to enhance the performance of social recommendation with graph neural network techniques, In particular, we provided a principled approach to flexible integrate multi-graph (user-item graph, user social graph, and item-item graph). In addition, we introduced the viewpoint mechanism to distinguish the importance among users and items. Experimental results on two real-world datasets show that VMRec can outperform state-of-the-art baselines.

References

1. Berg, R., Kipf, T.N., Welling, M.: Graph convolutional matrix completion. arXiv preprint arXiv:1706.02263 (2017)
2. Fan, W., et al.: Graph neural networks for social recommendation. In: The World Wide Web Conference (2019)
3. Fan, W., Ma, Y., Yin, D., Wang, J., Tang, J., Li, Q.: Deep social collaborative filtering. In: ACM RecSys (2019)
4. Guo, G., Zhang, J., Yorke-Smith, N.: TrustSVD: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In: Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
5. Jiang, M., Cui, P., Wang, F., Zhu, W., Yang, S.: Scalable recommendation with social contextual information. IEEE TKDE **26**, 2789–2802 (2014)
6. Rendle, S.: Factorization machines. In: 2010 IEEE International Conference on Data Mining, pp. 995–1000. IEEE (2010)
7. Tang, J., Chang, Y., Liu, H.: Mining social media with social theories: a survey. SIGKDD Explor. Newsl. **15**, 20–29 (2014)
8. Wu, L., Sun, P., Fu, Y., Hong, R., Wang, X., Wang, M.: A neural influence diffusion model for social recommendation. In: ACM SIGIR (2019)
9. Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J.: Graph convolutional neural networks for web-scale recommender systems. In: KDD (2018)