Hierarchical Graph Neural Networks for Personalized Recommendations with User-Session Context

Xiang Shen[1](✉), Caiming Yang[1], Zhengwei Jiang[2], Dong Xie[1],
Yingtao Sun[1], Shuibiao Chen[1], and Bo Li[3]

[1] State Grid Shaoxing Electric Power Supply Company,
Shaoxing, Zhejiang, China
shx8022@163.com.cn, 13606751852@163.com.cn,
13606755005@163.com, 416024740@qq.com,
13957569100@163.com

[2] State-Grid Zhejiang Electric Power Co., Ltd., Beijing, China
jiang_zhengwei@zj.sgcc.com.cn

[3] Beihang University, Beijing, China
libo@act.buaa.edu.cn

Abstract. With the swift development of Internet and artificial intelligence, recommender systems have become more and more important as useful information has a risk of submerging in huge amounts of data or smart services always provide a behavior predictor to meet diversified user's needs. User-session based recommendations are commonly applied in many modern online platforms. Graph Neural Networks (GNNs) have been shown to have a strong ability to address the problem of session-based recommendation with accurate item embedding. However, there are a lot of application scenarios that have already provided user profiles. We propose a model based on Hierarchical GNNs for personalized recommendations, which evolves both item information in sessions and user profiles. The experiments on two industry datasets show the superiority of our model over the state-of-the-art methods.

Keywords: Graph Neural Networks · User-session based recommendation · Dropout · Personalized recommendation

1 Introduction

With a dramatic increase in the Web information environment, information overload has received wide social attention. In most of all online services, recommender modules have become essential components to help users relieve the pressure on information overload and obtaining interesting information from huge amounts of data. Recommender systems can be also used to increase personal comfort by suppling items or products orderly with gauge of historical activities or synergetic relationships of items or users. For example, recommender systems could automatically list items of interest, or give a suggestion of new discounts relevant to each other. Many recommender systems are working on the assumption that user identification, item label and past activities are

M. Qiu (Ed.): SmartCom 2019, LNCS 11910, pp. 341–348, 2019.
https://doi.org/10.1007/978-3-030-34139-8_34

available with the temporal variation. It is more importance to consider a framework with complex scenarios of information scarcity. The core and the most crucial link of recommender systems is to search a way of computing relevance degree of items, users, or user-items with or without temporal information. Temporal information is important as it could supply an evidence of when users take items into account. There is a fact that to compare with the items considered long ago, recent considered items are more likely to be taken. The temporal information with user's activity are recorded as sessions. A session is a group of interaction occurred in a given time frame. The time frame can be an hour, a day, several days, several weeks, even months or years.

Traditional methods of personalized recommendations usually provide functions by building up a user profile, such as user-user similarity based on past activities computed by collaborative filtering approaches [1] or latent factor vectors built by matrix factorization [2, 3]. There exists a problem of cold-start for recommendation approaches that need user past activities, and it is not rare that new accounts, unlogged users, or users without tracking information use recommender systems. Conventional recommender algorithms [4] have been applied to address the challenging problem of recommendations on session-based recommenders. RNNs (Recurrent Neural Networks) have been majority approaches with promising results in the research of session-based recommender systems [5–8]. There are some limitations in these aforementioned approaches, such as estimation of user representations without adequate user behavior in a session, complex transition among the distant items. To overcome the above limitations, SR-GNNs (Session-based Recommendation with Graph Neural Networks) [9] have been introduced to generate representations for graphs, which can be constructed directly from historical session sequences. Based on the session graph, it is easy to capture complex item transitions and make item embedding vectors with high quality. However, there is shortcoming that the user behavior in past session is non accessible, which might miss valuable information for the present session, in SR-GNNs.

In this paper, we provide a Hierarchical GNNs model for personalized recommendations, in which there are two layers of GNNs constructed to address the problems of both session-based recommenders and session-aware recommenders. The first layer of GNNs provides user identifiers and propagate information, which might give an improvement of recommendation accuracy, and the second layer of GNNs generates the recommendations coupled with the first layer. Extensive experiments conducted on two industry datasets demonstrate the effectiveness of our proposed approach over the state-of-arts.

2 Related Work

In this section, we review some related works on session-based recommender systems, including RNNs-based methods and GNNs-based methods.

Recurrent Neural Models. RNNs are often applied for capturing the feature of sequential data [10], which include image and video captioning, time series prediction and much more. Long Short-Term Memory (LSTM) [11] networks and Gated Recurrent Units (GRUs) [12] are two types of RNNs with powerful performance on various abovementioned tasks. In the work of [5], RNNs are first applied to session-based recommendations, which provide recommendations for new sessions by learning a one-hot representation from the clicked item-IDs in the current session. After that, several valuable works are proposed based on RNNs methods, such as proper data augmentation techniques [6], the fusion of sequential patterns and co-occurrence signals [7], and attention mechanism [8].

Graph Neural Networks. Neural networks have been exploited for capturing representation for graph-structured data, such as social network, knowledge bases and etc. In work of [13, 14], GNNs are proposed to operate on directed graphs by using a form of RNNs. GNNs have been applied for various tasks, such as script even prediction [15], situation recognition [16], and image classification [17]. Recently, SR-GNNs [9] have been employed efficiently for session-based recommendations. Different with [9], both session-based recommenders and session-aware recommenders are considered by applying a Hierarchical GNNs model.

3 The Proposed Method

In this section, we will provide the idea of our proposed Hierarchical GNN model, which consist of two layers of GNNs for personalized recommendation with user-context. Our model is based on SR-GNN model presented in [9], which aims to predict which item a user will click next based on current sequential session data. Different from SR-GNN, our work takes the evolution of the user interests over time into account, showing in Fig. 1. We describe our model by two parts, i.e. the architecture of hierarchical GNNs and learning process of supper parameters.

3.1 Architecture

As shown in Fig. 1, our HGNN model takes a user-level GNN (U-GNN) to model the user activity across sessions, and a session-level GNN (S-GNN) to model the global preference of common items from all sessions. Different from SR-GNN, at each time step, S-GNN and U-GNN make a joint decision to the result of computing recommendations. To SR-GNN, it is easier for our HGNN model to learn the user's preference on selection of item sequences. In [9], authors claimed that sessions could be represented directly by nodes involved in a session, then they focused on developing a method to learn long-term preference and current interests on items from sessions. In their situation, there doesn't exist the information of user activities across sessions. On the contrary, we have taken this factor into account, as there are a lot of information that user's based activities could reveal. For example, it is not hard to associate animations for toddlers with children, rather than a video of hot-pot making. However, the baby A may prefer Spider-Man than Superman, just contrary to the baby B. Therefore, if there are videos of Spider-Man and Superman simultaneously, the recommendation ranking of videos should be different for two babies.

In session-based recommendation, $V = \{v_1, v_2, \ldots, v_m\}$ stands for the set consisting of all unique items involved in all the sessions. The clicked sequence S_u is ordered by

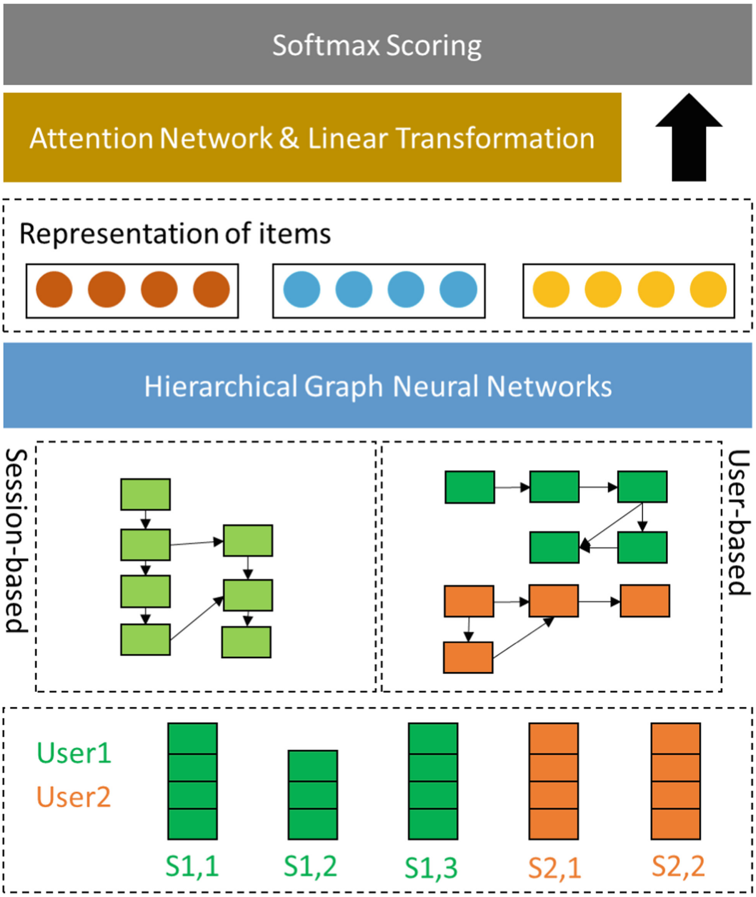

Fig. 1. Generic structure and workflow of our model.

timestamps from a session of the user U. In order to predict the next click, top-K values will be the basis of the candidate selection for recommendation. Same to [9], we embed every item into a unified embedding space and the latent vector of each item is learned via GNNs, and based on node vectors, each session can be represented by two embedding vector, which are composed of node vectors used in U-GNNs and S-GNNs.

3.2 Learning

In our S-GNNs, we take the same learning method for item embedding, and it is similar to obtain update functions for U-GNNs. It is different to obtain long-term preference and current interests of the input session after feeding all session graphs into GNNs. As in the workflow of our model, U-GNNs and S-GNNs will generate two independent local or global embedding vectors. To compute the hybrid embedding vector by taking linear transformation over the concatenation of four embedding vectors, including local and global embedding vectors obtained from U-GNNs and S-GNNs.

From our framework, when user is unknown our model would deteriorate to SR-GNN model, as the anonymous user makes U-GNNs same to S-GNNs. And we will test this part for our model. We applied the Back-Propagation through Time (BPTT) algorithm to train our model.

4 Experiments

In this section we describe our experimental setup and provide an in-depth discussion of the achieved results.

4.1 Datasets

Same to [9], we also evaluate the proposed method on two public representative datasets, including Yoochoose [18] and Diginetica [19]. The Yoochoose and Diginetica datasets are obtained from the RecSys Challenge 2015 and CIKM Cup 2016 respectively. The first dataset consists of a six-month stream of user clicks on an e-commerce website, and for the second one only transactional data is used.

As shown in Table 1, the most recent training sequences of Yoochoose are employed in the ratio of 1/128. In order to compare fairly with other methods, we drop sessions and items, which cannot meet session length greater than 1 or item occurrences not less than 5. Then there is a data-partition policy for generating training and test sets from the input sequence, with which the sessions of subsequent days and months (i.e. 30 days) are divided as test sets for Yoochoose and Diginetica respectively. Furthermore, to evaluate the performance of our approach, we filter out all anonymous users on the Diginetica dataset.

Table 1. The datasets used in experiments

	Yoochoose 1/64	Diginetica
Number of training sessions	184,929	184,701
Number of test sessions	55,898	6,473
Number of items	14,381	17,355
Number of users	NA	87,934
Average length	4.530	4.476

4.2 Evaluation metrics and parameter setup

There are two types of metrics taken for measuring performance of compared methods, including P@20 and MRR@20. These metrics are designed for the recommendation setting, as is often to make multiple recommendations for each user. P@20 (Precision), which is a popular measure of predictive accuracy, stands for the proportion of correctly recommended items amongst the top-20 items. MRR@20 (Mean Reciprocal

Rank), as the average of reciprocal ranks of correctly recommended items, takes the order of ranking into account. The number 20 means to set MRR to 0 once ranking exceeds 20. The larger the MRR value, the higher the ranking of correct recommendations in the top list.

To evaluate the performance of our proposed method, there are five representative baselines involved to comparisons, including POP, User-KNN, GRU4REC [5], NARM [20] and SR-GNN [9]. POP and S-POP recommend the top-N frequent items in the training set and in the current session respectively. GRU4REC employs RNNs to provide session-based recommendations by modeling the session sequences. NARM also applies RNNs to make recommendations, however it take attention mechanism into account for capturing user's sequential behavior and main purpose. SR-GNN exploits GNNs to model session sequences seamless with the ability of computing complex transitions of items.

Following previous method [9], the dimensionality of latent vectors $d = 100$ is established for both datasets. We initialize all parameters with a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. In order to optimize these parameters, a mini-batch Adam optimizer is applied with the initial learning rate of 0.001. And we set the L2 penalty and the batch size to 10^{-5} and 100 respectively.

4.3 Comparison with other methods

We evaluate the performance of our proposed approach by making comparisons with several other session-based recommendation methods, where the result in terms of P@20 and MRR@20 is listed in Table 2.

Table 2. Comparisons of various methods over three datasets

	Yoochoose 1/128		Diginetica	
	P@20	MRR@20	P@20	MRR@20
POP	6.64	1.59	0.87	0.18
User-KNN	–	–	37.81	12.31
GRU4REC	58.54	21.87	29.31	8.12
NARM	66.91	27.31	48.45	15.91
SR-GNN	68.71	29.96	49.46	17.29
Ours	**68.73**	**29.97**	**50.11**	**17.79**

From the result listed in Table 2, it is obvious that our proposed approach is outstanding from other five methods in terms of P@20 and MRR@20 over three datasets. In the architecture of our proposed model, we not only consider the impact of items of common sessions, but also take user's preference as well. All these session sequences are integrated as graph-structured data seamless. The result also shows that neural network-based approaches outperform the traditional methods by adopting deep learning mechanisms. Furthermore, SR-GNN is so outstanding from GRU4REC and NARM because of capturing more complex and implicit connections between user clicks. To be more advanced, our proposed model takes consider of user session information as well as global session information.

5 Conclusions and Future Work

In this paper, we proposed a mode based on Hierarchical GNNs to address the problem of personalized recommendations with user-session context, which exploits two-layer GNNs by involving both session-based recommenders and session-aware recommenders. This method is able to capture temporal information by analyzing the user activity over sessions and the present session status, which means that our proposed method has an ability to acquire the knowledge hidden in the long-term dynamics of user sessions. Our proposed Hierarchical GNNs model outperform state-of-the-art session-based methods for personalized recommendation by taking experiments on Diginetica.

In future works, we have a plan to search the way of refining user representations by exploiting features of items and users.

Acknowledgement. The authors gratefully acknowledge the anonymous reviewers for their helpful suggestions.

References

1. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted boltzmann machines for collaborative filtering. In: Proceedings of the 24th International Conference on Machine Learning, ICML (2007)
2. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining (2008)
3. Koren, Y., Bell, R.M., Volinsky, C.: Matrix factorization techniques for recommender systems. IEEE Comput. **42**(8), 30–37 (2009)
4. Koenigstein, N., Koren, Y.: Towards scalable and accurate item-oriented recommendations. In: Proceedings of the 7th ACM Conference on Recommender Systems (2013)
5. Hidasi, B., Karatzoglou, A., Baltrunas, L., et al.: Session-based recommendations with recurrent neural networks. In: ICLR (2016)
6. Tan, Y.K., Xu, X., Liu, Y.: Improved recurrent neural networks for session-based recommendations. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (2016)
7. Jannach, D., Ludewig. M.: When recurrent neural networks meet the neighborhood for session-based recommendation. In: Proceedings of the 11th ACM Conference on Recommender Systems (2017)
8. Li, J., Ren, P., Chen, Z., et al.: Neural attentive session-based recommendation. In: Proceedings of the 2017 ACM Conference on Information and Knowledge Management (2017)
9. Wu, S., Tang, Y., Zhu, Y., et al.: Session-based recommendation with graph neural networks. In: Proceedings of AAAI Conference on Artificial Intelligence (2019)

10. Lipton, Z.C., Berkowitz, J., Elkan, C.: A critical review of recurrent neural networks for sequence learning. CoRR, 1506.00019 (2015)
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
12. Cho, K., van Merrienboer, B., Bahdanau, D., et al.: On the properties of neural machine translation: encoder-decoder approaches. In: Proceedings of 8th Workshop on Syntax, Semantics and Structure in Statistical Translation (2014)
13. Gori, M., Monfardini, G., Scarselli, F.: A new model for learning in graph domains. In: IJCNN, vol. 2, pp. 729–734 (2005)
14. Scarselli, F., Gori, M., Tsoi, A.C., et al.: The graph neural network model. TNN **20**(1), 61–80 (2009)
15. Li, Z., Ding, X., Liu, T.: Constructing narrative event evolutionary graph for script event prediction. In: Proceedings of International Joint Conferences on Artificial Intelligence Organization (2018)
16. Li, R., Tapaswi, M., Liao, R., et al.: Situation recognition with graph neural networks. In: Proceedings of IEEE International Conference on Computer Vision (2017)
17. Marino, K., Salakhutdinov, R., Gupta, A.: The more you know: using knowledge graphs for image classification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2017)
18. Yoochoose. http://2015.recsyschallenge.com/challege.html
19. Diginetica. http://cikm2016.cs.iupui.edu/cikm-cup
20. Li, J., Ren, P., Chen, Z., et al.: Neural attentive session-based recommendation. In: CIKM (2017)