Spatio-Temporal Attentive Network for Session-Based Recommendation

Chunkai Zhang(✉) and Junli Nie

Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
ckzhang@hit.edu.cn, njl_primary@163.com

Abstract. Session-based recommendation aims to predict the user's next click behavior based on the existing anonymous session information. Existing methods either only utilize temporal information of the session to make recommendations or only capture complex item transitions from spatial perspective to recommend, they are insufficient to obtain rich item representations. Besides, user's real purpose of the session is also not emphasized. In this paper, we propose a novel session-based recommendation method, named Spatio-Temporal Attentive Session-based Recommendation, STASR for brevity. Specifically, we design a hybrid framework based on Graph Neural Network (GNN) and Gated Recurrent Unit (GRU) to obtain richer item representations from spatio-temporal perspective. During the process of constructing corresponding session graph in GNN, an individual-level skipping strategy, which considers the randomness of user's behaviors, is proposed to enrich item representations. Then we utilize attention mechanism to capture the user's real purpose involved user's initial will and main intention. Extensive experimental results on three real-world benchmark datasets show that STASR consistently outperforms state-of-the-art methods on a variety of common evaluation metrics.

Keywords: Session-based recommendation · Spatio-temporal perspective · Attention mechanism

1 Introduction

Recommendation systems help users alleviate the problem of information overload and suggest items that may be of interest to users. Traditional recommendation methods [1], such as content-based methods and collaborative filtering methods, utilize user profiles and user-item interaction records to recommend. However, in many services, such as e-commerce websites and most media sites, user profiles may be unknown, and only the on-going session is available. Under these circumstances, session-based recommendation [5] is proposed to predict user's next behavior based merely on the history click records in the current session.

G. Li et al. (Eds.): KSEM 2020, LNAI 12275, pp. 131–139, 2020.
https://doi.org/10.1007/978-3-030-55393-7_13

Recently, a lot of researches have begun to realize the importance of Recurrent Neural Network (RNN) for session-based recommendation. Hidasi et al. [2] first employ the Gated Recurrent Unit(GRU) to solve session-based recommendation problem. Then Tan et al. [6] improve the recommendation performance via considering data augmentation and temporal shifts of user behaviors. Recently, Li et al. [3] propose NARM based on encoder-decoder architecture to capture the sequence behavior and main purpose of the user simultaneously. In contrast to NARM, STAMP [4]aims to capture user's long-term and short-term interests to make effective recommendations. Although these RNN-based methods above have achieved significant results, they only consider single-way transitions between consecutive items and neglect the transitions among the contexts, i.e., other items in the session.

Graph Neural Network(GNN) [7] is designed to learn the representations for graph structured data. As for session-based recommendation, SRGNN [8] converts session into the form of session graph, then utilizes GNN to generate accurate item representations via capturing complex item transitions. Then Xu et al. [9] propose a graph contextualized self-attention network based on both GNN and self-attention to model local graph-structured dependencies of separated session sequences and obtain contextualized non-local representations.

Although the existing GNN-based methods achieve excellent performance, they still have some limitations. Firstly, they only emphasize complex item transitions of the current session from spatial perspective. They ignore the impact of repeated user behavior pairs on user's interests, i.e., session $[x_1, x_2, x_3, x_2, x_4]$ and session $[x_1, x_2, x_3, x_2, x_3, x_2, x_4]$ correspond to the same session graph as well as the same item representations, which confines the prediction accuracy. In other words, they neglect temporal information of the whole session. Secondly, the randomness of user's behaviors is ignored, we also call it as individual-level skip behaviors of the user in the current session, i.e., the past one behavior not only has direct impact on the next behavior but also may have direct impact on the behavior after skipping a few time steps. Thirdly, previous work does not emphasize the user's real purpose involved user's initial will and main intention of the current session.

To overcome these limitations, in this paper, we propose a novel method for session-based recommendation. The main contributions of our work can be summarized as:

- We design a hybrid framework based on GNN and GRU to obtain richer item representations from spatio-temporal perspective. During the process of constructing corresponding session graph in GNN, an individual-level skipping strategy, which considers the randomness of user's behaviors, is proposed to enrich item representations.
- We apply two attention networks to extract the user's real purpose involved user's initial will and main intention in the current session.
- We carried out extensive experiments on three real-world benchmark datasets. The results demonstrate that our proposed method performs better than state-of-art methods in terms of Recall@20 and MRR@20.

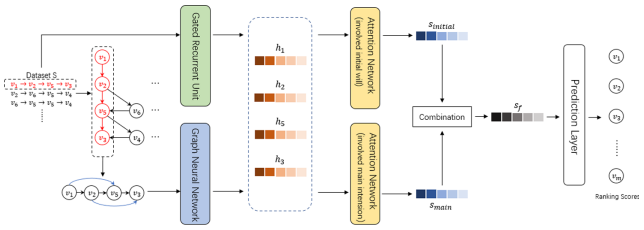

Fig. 1. The graphical model of our proposed method.

2 Related Work

Hidasi et al. [2] first design a RNN model with GRU to predict user's next behavior based on previous behaviors in the current session. Then Tan et al. [6] propose improved version of GRU4Rec to boost performance via two methods, i.e., data augmentation and accounting for shifts in the input data distribution. Recently, NARM [3] takes advantage of an encoder-decoder architecture to consider sequence behavior features and capture main purpose of the current session simultaneously. Then STAMP [4] using MLP networks and attention mechanism, is proposed to efficiently capture both the user's long-term and short-term interests of a session. SRGNN [8] is first proposed to convert the session into the form of session graph, and uses GNN to capture more complex items transitions based on the session graph. Then Xu et al. [9] propose a graph contextualized self-attention network to capture local graph-structured dependencies and contextualized non-local representations simultaneously.

3 Method

3.1 Notations

Let $V = \{v_1, v_2, ..., v_{m-1}, v_m\}$ represents the set of unique items appearing in the whole dataset. Anonymous session is denoted as $S = [x_1, x_2, ..., x_{n-1}, x_n]$, where $x_t \in V(1 \leq t \leq n)$ denotes the item clicked at time step t. As for any given prefix of session $[x_1, x_2, ..., x_t](1 \leq t \leq n)$, our proposed method aims to model the current session and predict the user's next behavior x_{t+1}. In many online services, recommendation systems provide a ranking list $y = [y_1, y_2, ..., y_{m-1}, y_m]$ over all candidate items for the user, where $y_j(1 \leq j \leq m)$ represents the probability of item j clicked by the user at the next time step.

3.2 Item Representation Layer

– First, we utilize GNN with an individual-level skipping strategy to obtain item representations from spatial perspective, which can capture complex item transitions and consider the randomness of user behaviors.

Construct session graph. The first part of GNN is to construct corresponding session graph. Different from the way of constructing the directed session graph in SRGNN [8], we propose an individual-level skipping strategy to consider the randomness of user behaviors (i.e., skip behaviors in the current session). To improve the robustness and keep the simplicity of our model, the individual-level skipping strategy is conducted via adding directed connections according to chronological order between two items randomly at a certain ratio in the corresponding session graph. $M^O, M^I \in \mathbb{R}^{n \times n}$ represent weighted connections of outgoing and incoming edges in the session graph. For example, given a session $S = [v_1, v_3, v_2, v_4, v_5, v_6, v_2]$, to consider the randomness of user behaviors, we apply skipping strategy to obtain corresponding session graph and matrices M^O, M^I, which are shown in Fig. 2. Following previous work [8], since some items maybe appear repeatedly in a session, the normalization process is often used when constructing the outgoing matrix and incoming matrix, which is calculated as the occurrence of the edge divided by the outdegree of that edge's start node.

Node representation learning. Here, we describe how to update node representations based on the constructed session graph and matrices M^O, M^I, as for each node in the session graph, its update functions are given as follows:

$$a_v^{(t)} = A_{v:}^T [h_1^{(t-1)T} ... h_{|V|}^{(t-1)T}]^T + b \tag{1}$$

$$z_v^t = \sigma(W^z a_v^{(t)} + U^z h_v^{(t-1)}) \tag{2}$$

$$r_v^t = \sigma(W^r a_v^{(t)} + U^r h_v^{(t-1)}) \tag{3}$$

$$\tilde{h}_v^{(t)} = tanh(W a_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \tag{4}$$

$$h_v^{(t)} = (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \tilde{h}_v^{(t)} \tag{5}$$

$A_{v:}$ is defined as the combination of the two columns corresponding to node v from the outgoing and incoming matrices. $a_v^{(t)}$ extracts the contextual information of adjacent nodes for node v. z_v^t and r_v^t are update gate and reset gate respectively. $\tilde{h}_v^{(t)}$ represents the newly generated information, and $h_v^{(t)}$ is the final updated node state. In addition, $\sigma(\cdot)$ denotes the logistic sigmoid function and $\odot$ denotes element-wise multiplication.

– Then, we utilize GRU to consider temporal information including repeated user behavior pairs of the current session into account. GRU is more simplified than the standard RNN and its update formulas are as follows:

$$z_t = \sigma(W_z \cdot [h_{t-1}, v_t]) \tag{6}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, v_t]) \tag{7}$$

$$\tilde{h_t} = tanh(W \cdot [r_t * h_{t-1}, v_t]) \tag{8}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h_t} \tag{9}$$

z_t,r_t represent update gate and reset gate respectively. And h_t denotes the activation of GRU which is a linear interpolation between the previous activation h_{t-1} and the candidate activation $\tilde{h_t}$. Here, we essentially use the

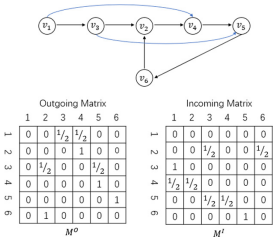

Fig. 2. An example of session graph structure and connection matrices M^O, M^I after applying individual-level skipping strategy.

corresponding hidden states $[h_1, h_2, ..., h_{t-1}, h_t]$ of the input as the item representations of the current session sequence from temporal perspective.

Finally, we could obtain richer item representations from spatio-temporal perspective, which are combined as the unified item representations later.

3.3 Attention Layer

Here, we apply two item-level attention networks to dynamically choose more important items and linearly combine each part of the input for the user's initial will and main intention respectively. The formulas are defined as follows:

$$s_{initial} = \sum_{j=1}^{t} \alpha_{tj} h_j \tag{10}$$

$$s_{main} = \sum_{j=1}^{t} \beta_{tj} h_j \tag{11}$$

where

$$\alpha_{tj} = v^T \sigma(W_1 h_1 + W_2 h_j + c_1) \tag{12}$$

$$\beta_{tj} = q^T \sigma(W_3 h_t + W_4 h_j + c_2) \tag{13}$$

α_{tj} and β_{tj} determine the importance of each item in the session when we consider user's initial will and main intention respectively. In detail, α_{tj} is used to compute the similarity between h_1 and the representation of previous item h_j. And β_{tj} computes the similarity between the final item representation h_t and the representation of previous item h_j. $\sigma(\cdot)$ is an activate function and matrices W_1, W_2, W_3, W_4 control the weights. Finally, we obtain the session representation s_f by taking linear transformation over the concatenation of $s_{initial}$ and s_{main}.

$$s_f = W_5[s_{initial}; s_{main}] \tag{14}$$

Matrix W_5 is used to compress these two combined embedding vectors into the latent space.

3.4 Prediction Layer

Here we calculate corresponding probability of each candidate item v_i being clicked at the next time step. The computing formula can be defined as:

$$\hat{z_i} = {s_f}^T v_i \tag{15}$$

Then we apply a softmax function to get the output vector of the model:

$$\hat{y} = softmax(\hat{z}) \tag{16}$$

For each session, its loss function can be defined as cross-entropy of the prediction and the ground truth:

$$L(\hat{y}) = -\sum_{i=1}^{m} y_i log(\hat{y_i}) + (1 - y_i) log(1 - \hat{y_i}) \tag{17}$$

y_i denotes the one-hot encoding vector of the ground truth item.

Finally, our proposed model is trained by Back-Propagation Through Time (BPTT) algorithm in the learning process.

Table 1. Statistics of the datasets used in our experiments

Datasets	#clicks	#train	#test	#items	avg.length
Diginetica	982961	719470	60858	43097	5.12
Yoochoose1/64	557248	369859	55898	16766	6.16
Retailrocket	710856	433648	15132	36968	5.43

4 Experiments and Analysis

4.1 Settings

Datasets. Yoochoose is a public dataset released on RecSys Challenge 2015. Diginetica is obtained from CIKM Cup 2016 competition, in our experiment, we only use the click records dataset. Retailrocket comes from an e-commerce company, we select the user's browsing history records dataset in the experiment. We also filter out sessions with length of 1 and items appearing less than 5 times in all datasets. For Yoochoose dataset, we select the sessions from the last day as the test set and the other as the training set, for the Diginetica and Retailrocket datasets, we select the sessions from the last week as the test set and the others as the training set. The statistics of the datasets is shown in Table 1.

Evaluation Metrics. Recall is an evaluation of unranked retrieval results, which represents the proportion of correctly recommended items among Top-N items. MRR(Mean Reciprocal Rank) is an evaluation metric of ranked list, which indicates the correct recommendations in the Top-N ranking list.

Baselines. To show the effectiveness of our proposed method, we compare it with six methods: POP, S-POP, GRU4Rec, NARM, STAMP and SRGNN. POP and S-POP are traditional recommendation methods. GRU4Rec, NARM and STAMP are RNN-based methods. SRGNN is a session-based recommendation method with GNN.

Parameter Setup. We initially set the dimensionality of latent vectors as 160 on Yoochoose dataset and 100 on Diginetica and Retailrocket datasets. All parameters are initialized using a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. The initial learning rate is set to 0.001 and will decay by 0.1 after every 3 epochs. The number of epochs is set to 30 and 10% of the training data is used as validation set.

4.2 Comparison Results

The results of all methods over three real-world datasets in terms of Recall@20 and MRR@20 are shown in Table 2.

Table 2. Comparison of our proposed method with baseline methods over three real-world datasets

Methods	Diginetica		Yoochoose1/64		Retailrocket	
Measures	Recall@20	MRR@20	Recall@20	MRR@20	Recall@20	MRR@20
POP	0.89	0.2	6.71	1.65	1.24	0.32
S-POP	21.06	13.68	30.44	18.35	40.48	32.04
GRU4Rec	29.45	8.33	60.64	22.89	55.59	32.27
NARM	49.7	16.17	68.32	28.63	61.79	34.07
STAMP	45.64	14.32	68.74	29.67	61.08	33.1
SRGNN	50.73	17.59	70.57	30.94	62.79	34.49
Ours	**52.58**	**18.28**	**71.32**	**31.01**	**64.15**	**35.23**

We have the observation from the results that compared to all the baseline methods, our proposed method achieves better performance among all the methods on three real-world datasets in terms of Recall@20 and MRR@20.

4.3 Model Analysis and Discussion

To verify the performance of different components in our model, we conduct a series of experiments. The results are shown in Table 3, we can observe that the hybrid framework plays an important role in recommending results. A possible reason is that we make recommendations based on item representations, so it is important to obtain rich item representations considering various user's behaviors from different perspectives. From Table 3, we can also observe that only considering a single feature, i.e., user's initial will or main intention, does not perform better than considering both features.

Table 3. The performance of our proposed method with and without different components in terms of Recall@20 and MRR@20.

Methods	Diginetica		Yoochoose1/64		Retailrocket	
Measures	Recall@20	MRR@20	Recall@20	MRR@20	Recall@20	MRR@20
w/ hybrid framework	52.58	18.28	71.32	31.01	64.15	35.23
w/o hybrid framework	51.59	17.93	70.68	30.77	63.09	34.59
only w/ main intention	52.36	18.04	70.79	30.86	63.31	34.94
only w/ initial will	52.31	17.96	70.80	30.84	63.27	34.65

5 Conclusion

In this paper, we propose a novel method named STASR for session-based recommendation. Specifically, we design a hybrid framework based on GNN with individual-level skipping strategy and GRU to obtain richer item representations from spatio-temporal perspective. Besides, user's real purpose involved user's initial will and main intention is considered for accurate recommendation. On three datasets, our proposed method can consistently outperform other state-of-art methods.

Acknowledgments. This work was supported by Natural Science Foundation of Guangdong Province, China (Grant NO.2020A1515010970) and Shenzhen Research Council (Grant NO.GJHZ20180928155209705).

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **6**, 734–749 (2005)
2. Hidasi, B., Karatzoglou, A., Baltrunas, L., Tikk, D.: Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939 (2015)
3. Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T., Ma, J.: Neural attentive session-based recommendation. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1419–1428. ACM (2017)
4. Liu, Q., Zeng, Y., Mokhosi, R., Zhang, H.: Stamp: short-term attention/memory priority model for session-based recommendation. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1831–1839. ACM (2018)
5. Schafer, J.B., Konstan, J., Riedl, J.: Recommender systems in e-commerce. In: Proceedings of the 1st ACM Conference on Electronic Commerce, pp. 158–166. ACM (1999)
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, pp. 17–22. ACM (2016)
7. Wang, X., He, X., Wang, M., Feng, F., Chua, T.S.: Neural graph collaborative filtering. arXiv preprint arXiv:1905.08108 (2019)

8. Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., Tan, T.: Session-based recommendation with graph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 346–353 (2019)
9. Xu, C., Zhao, P., Liu, Y., Sheng, V.S., Xu, J., Zhuang, F., Fang, J., Zhou, X.: Graph contextualized self-attention network for session-based recommendation. In: Proceedings of 28th International Joint Conference on Artificial Intelligence (IJCAI), pp. 3940–3946 (2019)