Attention with Long-term Interval-Based Gated Recurrent Units for Modeling Sequential User Behaviors

Zhao Li[1],Chenyi Lei[1],Pengcheng Zou[1],Donghui Ding[1], Shichang Hu[1],Zehong Hu[1],Shouling Ji[2],Jianliang Gao[3]

[1] Alibaba-Group, China
{lizhao.lz,chenyi.lcy,xuanwei.zpc,donghui.ddh,shichang.hsc,zehong.hzh}@alibaba-inc.com
[2] Zhejiang University, Hangzhou, Zhejiang, China
sji@zju.edu.cn
[3] Central South University, China
gaojianliang@csu.edu.cn

Abstract. Modeling user behaviors as sequences provides key advantages in predicting future user actions for personalized recommendation. To utilize sequential user behaviors effectively, traditional methods usually depend on the premise of Markov processes, and recurrent neural networks (RNNs) have been adopted to leverage their power in modeling sequences recently. In this paper, we design a network featuring **A**ttention with **L**ong-term **I**nterval-based **G**ated **R**ecurrent **U**nits (ALI-GRU) to model temporal sequences of user actions. Compared to the existing methods, our network utilizes the time interval-based GRU in addition to normal GRU to exploit the temporal dimension when encoding user actions, and has a specially designed matrix-form attention function to characterize both long-term preferences and short-term intents of users. The attention-weighted features are finally decoded to predict the next user action. We have performed experiments on well-known public datasets. Experimental results show that the proposed ALI-GRU achieves significant improvement than state-of-the-art RNN-based methods.

Keywords: Attention Mechanism · Recurrent Neural Networks · User Modeling.

1 Introduction

Traditional personalized recommendation methods, such as item to item collaborative filtering did not consider the dynamics of user behaviors. For example, to predict user's next action such as the next product to purchase, the profiling of both long-term preferences and short-term intents of user are required. Therefore, modeling the user's behaviors as sequences provides key advantages. Nonetheless, modeling sequential user behaviors raises even more challenges than modeling them without the temporal dimension. How to identify the correlation and dependence among actions is one of the difficult issues. Recently, many different kinds of RNN algorithms have been proposed for modeling user behaviors to leverage their powerful descriptive ability for sequential data [6, 9]. However, there are several limitations that make it difficult to apply these methods into the wide variety of applications in the real-world. One inherent assumption of these methods is that the importance of historical behaviors decreases over time, which is also the intrinsic property of RNN cells such as gated recurrent units (GRU) and long- and short-term memory (LSTM). This assumption does not always apply in practice, where the sequences may have complex cross-dependence. In this paper, we propose a network featuring **A**ttention with **L**ong-term **I**nterval-based **G**ated **R**ecurrent **U**nits (ALI-GRU) for modeling sequential user behaviors to predict user's next action. We adopt a series of bi-directional GRU to process the sequence of items that user had accessed. The GRU cells in our network consist of time interval-based GRU, where the latter is to reflect the short-term information of time intervals. In addition, the features extracted by bi-directional GRU are used to drive an attention model, where the attention distribution is calculated at each timestamp. We have performed a series of experiments using well-known public datasets. Experimental results show that ALI-GRU outperforms the state-of-the-art methods by a significant margin.

2 Related Work

The related work is given at two aspects, modeling of sequential user behaviors and attention mechanism.

Modeling Sequential User Behaviors Due to the significance to user-centric tasks such as personalized search and recommendation, modeling sequential user behaviors has attracted great attention from both industry and academia. Most of pioneering work relies on model-based Collaborative Filtering (CF) to analyze user-item interaction matrix. For the task of sequential recommendation, Rendle *et al.* [11] propose Factorizing Personalized Markov Chain to combine matrix factorization of user-item matrix with Markov chains. He *et al.* [4] further integrate similarity-based methods [8] into FPMC to tackle the problem of sequential dynamics. But the major problems are that these methods independently combine several components, rely on low-level hand-crafted features of user or item, and have difficulty to handle long-term behaviors. To the contrary, with the success of recurrent neural networks (RNNs) in the past few years, a paucity of work has made attempts to utilize RNNs [5]. The insight that RNN-based solutions achieve success in modeling sequential user behaviors is that the well demonstrated ability of RNN in capturing patterns in the sequential data. Recent studies [10, 15, 12] also indicate that time intervals within sequential signal are a very important clue to update and forget information in RNN architecture. But in practice, there is complex dependence and correlation between sequential user behaviors, which requires deeper analysis of relation among behaviors rather than simply modeling the presence, order and time intervals. To summarize, how to design an effective RNN architecture to model sequential user behaviors effectively is still a challenging open problem.

Attention Mechanism The success of attention mechanism is mainly due to the reasonable assumption that human beings do not tend to process the entire signal at once, but only focus on selected portions of the entire perception space when and where needed [7]. Recent researches start to leverage different attention architectures to improve performance of related tasks. For example, Yang *et al.* [14] propose a hierarchical attention network at word and sentence level, respectively, to capture contributions of different parts of a document. Vaswani *et al.* [13] utilize multi-head attention mechanism to improve performance. Nevertheless, most of previous work calculates attention distribution according to the interaction of every source vector with a single embedding vector of contextual or historical information, which may lead to information loss caused by early summarization, and noise caused by incorrect previous attention.

Indeed, the attention mechanism is very sound for the task of modeling sequential user behaviors. However, to the best of our knowledge, there is few work concentrating on this paradigm, except a recent study [1], which considers the attention mechanism into a multimedia recommendation task with multilayer perceptron. An effective solution with attention mechanism for better modeling sequential user behaviors is to be investigated in this paper.

3 ALI-GRU

We start our discussion with some definition of notations. Let $\mathcal{U}$ be a set of users and $\mathcal{I}$ be a set of items in a specific service such as products in online shopping websites. For each user $u \in \mathcal{U}$, his/her historical behaviors are given by $\mathcal{H}^u = \{(i_k^u, t_k^u) | i_k^u \in \mathcal{I}, t_k^u \in \mathcal{R}^+, k = 1, 2, \ldots, N_u\}$, where (i_k^u, t_k^u) denotes the interaction between user u and item i_k^u at time t_k^u, interaction has different forms in different services, such as clicking, browsing, adding to favorites, etc. The objective of modeling sequential user behaviors is to predict the conditional probability of the user's next action $p(i_{N_u+1}^u | \mathcal{H}^u, t_{N_u+1}^u)$ for a certain given user u.

As illustrated in the left part of Fig. 1, our designed network features an attention mechanism with long-term interval-based gated recurrent units for modeling sequential user behaviors. This network architecture takes the sequence of items as raw signal. There are four stages in our network. The embedding layer maps items to a vector space to extract their basic features. The bi-directional GRU layer is designed to capture the information of both long-term preferences and short-term intents of user, it consists of normal GRUs and time interval-based GRUs. The attention function layer reflects our carefully designed attention mechanism, which is illustrated in the right part of Fig. 1. Finally, there is an output layer to integrate the attention distribution and the extracted sequential features, and utilize normal GRUs to predict the conditional probability of next item.

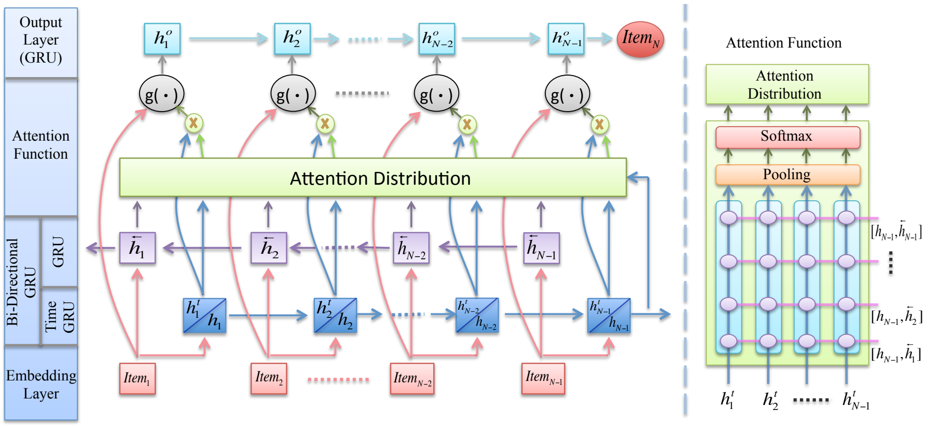

Fig. 1. (Best view in color). The proposed framework for modeling sequential user behaviors (left) and the designed attention mechanism (right).

Bi-directional GRU Layer with Time-GRU. This layer is designed to extract driven signals from input sequence and to refine the long-term memory by contextual cues. We propose a network structure with time-GRU to extract short-term dynamics of user intents as driven signal of the attention function.

The structure of time-GRU is different from the normal GRU. For the input I_N, the normal GRU computes linear interpolation between the last state h_{N-1} and the candidate activation $\tilde{h}_N$,

$$h_N = (1 - z_N) \odot h_{N-1} + z_N \odot \tilde{h}_N \tag{1}$$

where $\odot$ is an element-wise multiplication.

Since GRU is originally designed for NLP tasks, there is no consideration of time intervals within inputs. To include the short-term information, we augment the normal GRU with a time gate T_N

$$\begin{aligned} T_N =& \sigma(W_t \triangle t_N + U_t I_N + b_t) \\ &\text{s.t. } W_t < 0 \end{aligned} \tag{2}$$

where $\triangle t_N$ is the time interval between adjacent actions. The constraint $W_t < 0$ is to utilize the simple assumption that smaller time interval indicates larger correlation. Moreover, we generate a time-dependent hidden state h_N^t in addition to the normal hidden state h_N, i.e.

$$h_N^t = (1 - z_N \odot T_N) \odot h_{N-1}^t + z_N \odot T_N \odot \tilde{h}_N^t \tag{3}$$

where we utilize the time gate as a filter to modify the update gate z_N so as to capture short-term information more effectively.

In addition, to utilize contextual cues to extract long-term information, we propose to combine the output of forward normal GRU (h_N in Eq. (1)) with all the outputs of backward GRU at different steps (the output of backward GRU at step k is denoted by $\overleftarrow{h}_k$ in Fig. 1). Specifically, we produce concatenated vectors $[h_{N-1}, \overleftarrow{h}_{N-1}]$, $[h_{N-1}, \overleftarrow{h}_{N-2}]$, $\dots$, $[h_{N-1}, \overleftarrow{h}_1]$, as shown in the right part of Fig. 1, where $[,]$ stands for concatenation of vectors. This design effectively captures the contextual cues as much as possible.

Attention Function Layer. Unlike previous attention mechanisms, we do not simply summarize the contextual long-term information into individual feature vectors. We design to attend the driven signals at each time step along with the embedding of contextual cues.

Specifically, as shown in the right part of Fig. 1, we use $\mathbf{H}_k = [h_{N-1}, \overleftarrow{h}_k] \in \mathcal{R}^{2d}, k = 1, 2, \dots, N-1$, where d is the dimension of GRU states, to represent the contextual long-term information. $h_k^t \in \mathcal{R}^d$ denotes the short-term intent reflected by item i_k. We then construct an attention matrix $A \in \mathcal{R}^{(N-1)*(N-1)}$, whose elements are calculated by

$$A_{ij} = \alpha(\mathbf{H}_i, h_j^t) \in \mathcal{R} \tag{4}$$

where the attention weight

$$\alpha(\mathbf{H}_i, h_j^t) = v^T \tanh(W_a \mathbf{H}_i + U_a h_j^t) \tag{5}$$

is adopted to encode the two input vectors. There is a pooling layer along the direction of long-term information, and then a softmax layer to normalize the attention weights of each driven signal. Let a_k be the normalized weight on h_k^t, then the attended short-term intent vector is $\hat{h}_k^t = a_k h_k^t \in \mathcal{R}^d$. At last, we use $g(i_k, \hat{h}_k^t) = [i_k, \hat{h}_k^t, |i_k - \hat{h}_k^t|, i_k \odot \hat{h}_k^t] \in \mathcal{R}^{4d}$ as the output to the next layer, where i_k is the embedded vector of the item at the k-th step.

Our carefully designed attention mechanism described above is to reduce the loss of contextual information caused by early summarization. Furthermore, since driven signals are attended to the long-term information at different steps, the attentions can obtain the trending change of user's preferences, being more robust and less affected by the noise in the historical actions.

4 Experiments

To verify the performance of ALI-GRU, we conduct a series of experiments on two well-known public datasets (LastFM[4] and CiteULike[5]). We compare ALI-GRU with the following state-of-the-art approaches for performance evaluation: **Basic GRU/Basic LSTM [2]**,**Session RNN [5]**,**Time-LSTM [15].** All RNN-based models are implemented with TensorFlow. Training was done on a single GeForce Tesla P40 GPU with 8 GB graphical memory.

In this experiment, we use the datasets as adopted in [15], i.e. LastFM (987 users and 5000 items with 818767 interactions) and CiteULike (1625 users and

[4] http://www.dtic.upf.edu/∼ocelma/MusicRecommendationDataset/lastfm-1K.html

[5] http://www.citeulike.org/faq/data.adp

5000 items with 35834 interactions). Both datasets can be formulated as a series of tuples <user_id, item_id, timestamp>. Our target is to recommend songs in LastFM and papers in CiteULike for users according to their historical behaviors.

For the fair of comparison, we follow the segmentation of training set and test set in [15]. Specifically, 80% users are randomly selected for training. The remaining users are for testing. For each test user u with k historical behaviors, there are $k-1$ test cases, where the k-th test case is to perform recommendations at time t^u_{k+1} given the user's previous k actions, and the ground-truth is i^u_{k+1}. The recommendation can also be regarded as a multi-class classification problem.

We use one-hot representations of items as inputs to the network, and one fully-connected layer with 8 nodes for embedding. The length of hidden states of GRU-related layers including both normal GRU and Time-GRU is 16. A softmax function is used to generate the probability prediction of next items. For training, we use the AdaGrad [3] optimizer, which is a variant of Stochastic Gradient Descent (SGD). Parameters for training are minibatch size of 16 and initial learning rate of 0.001 for all layers. The training process takes about 8 hours.

Table 1. Recall@10 Comparison Results on LastFM & CiteULike

	LastFM	CiteULike
Basic-LSTM	0.2451	0.6824
Session-RNN	0.3405	0.7129
Time-LSTM	0.3990	0.7586
ALI-GRU	**0.4752**	**0.7764**

In the test stage, we use Recall@10 to measure whether the ground-truth item is in the recommendation list. The results of sequential recommendation tasks on LastFM and CiteULike are shown in Table. 1. It can be observed that our approach performs the best on both LastFM and CiteULike for all metrics, which demonstrates the effectiveness of our proposed ALI-GRU. Specifically, ALI-GRU obtains significant improvement over Time-LSTM, which is the best baseline, averagely by 10.7% for Recall@10. It owes to the superiority of introducing attention mechanism into RNN-based methods especially in capturing the contribution of each historical action.

Performance of Cold-start. Cold-start refers to the lacking of enough historical data for a specific user, which often decreases the efficiency of making recommendations. We analyze the influence of cold-start on the LastFM dataset and the results are given in Fig. 2. In this figure, test cases are separately counted for different numbers of historical actions, small number refers to cold-start. We can observe that for cold users with only 5 actions, ALI-GRU performs slightly worse than the state-of-the-art methods. This is because that ALI-GRU considers short-term information as driven signals, which averages source signal to some extent and leads to less accurate modeling for cold users. Along with the increase of historical actions, ALI-GRU achieves significantly better performance than the baselines, which indicates that bi-directional GRU and attention mechanism can better model the long-term preferences for making recommendations.

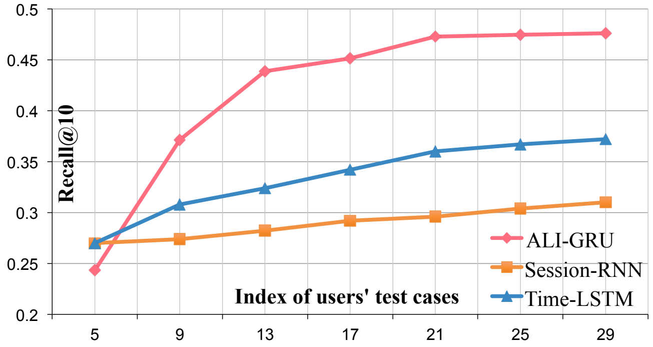

Fig. 2. Recall@10 evaluated on different indexes of users' test cases in LastFM.

5 Conclusions

In this paper, we propose to integrate a matrix-form attention mechanism into RNNs for better modeling sequential user behaviors. Specifically, we design a network featuring Attention with Long-term Interval-based Gated Recurrent Units to model temporal sequences of user actions, and using a Time-GRU structure to capture both long-term preferences and short-term intents of users as driven signals for better robustness. The empirical evaluations on two public datasets for sequential recommendation task show that our proposed approach achieves better performance than several state-of-the-art RNN-based solutions. One limitation of this work is the lack of user profiling in providing personalized content, which will be addressed in our future work.

Acknowledgment

This work was partly supported by the National Key Research and Development Program of China under No. 2018YFB0804102, NSFC under No. 61772466, U1936215, and U1836202, the Zhejiang Provincial Natural Science Foundation for Distinguished Young Scholars under No. LR19F020003, the Provincial Key Research and Development Program of Zhejiang, China under No. 2019C01055, the Ant Financial Research Funding, the Alibaba-ZJU Joint Research Institute of Frontier Technologies and Zhejiang Lab under No. 2019KE0AB01.

References

1. Chen, J., Zhang, H., He, X., Nie, L., Liu, W., Chua, T.: Attentive collaborative filtering: multimedia recommendation with reature- and item-level attention. In: Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 335–344. ACM (2017)
2. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NIPS Workshop on Deep Learning. MIT Press (2014)
3. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. In: Journal of Machine Learning Research. p. 2121–2159. ACM (February 2011)
4. He, R., McAuley, J.: Fusing similarity models with markov chains for sparse sequential recommendation. In: International Conference on Data Mining (ICDM). pp. 191–200. IEEE (2016)
5. Hidasi, B., Karatzoglou, A., Baltrunas, L., Tikk, D.: Session-based recommendations with recurrent neural networks. In: International Conference on Learning Representations. IEEE (2016)
6. Hidasi, B., Karatzoglou, A.: Recurrent neural networks with top-k gains for session-based recommendations. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. pp. 843–852 (2018)
7. Hubner, R., Steinhauser, M., Lehle, C.: A dual-stage two-phase model of selective attention. In: Psychological Review. pp. 759–784. APA (July 2010)
8. Kabbur, S., Ning, X., Karypis, G.: Fism: factored item similarity models for top-n recommender systems. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 659–667. ACM (2013)
9. Liu, Q., Wu, S., Wang, L.: Multi-behavioral sequential prediction with recurrent log-bilinear model. In: Transactions on Knowledge and Data Engineering. pp. 1254–1267. IEEE (June 2017)
10. Neil, D., Preiffer, M., Liu, S.: Phased lstm: accelerating recurrent network training for long or event-based sequences. In: Advances in neural information processing systems (NIPS). pp. 3882–3890. MIT Press (2016)
11. Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: Factorizing personalized markov chains for next-basket recommendation. In: Proceedings of the 19th International Conference on World Wide Web. pp. 811–820. ACM (2010)
12. Vassøy, B., Ruocco, M., de Souza da Silva, E., Aune, E.: Time is of the essence: A joint hierarchical rnn and point process model for time and item predictions. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 591–599 (2019)
13. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
14. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). pp. 1480–1489. NAACL (2016)
15. Zhu, Y., Li, H., Liao, Y., Wang, B., Guan, Z., Liu, H., Cai, D.: What to do next: modeling user behaviors by time-lstm. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 3602–3608. AAAI Press (2017)