IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 July 2024, accepted 3 August 2024, date of publication 8 August 2024, date of current version 16 August 2024.
Digital Object Identifier 10.1109/ACCESS.2024.3440351

RESEARCH ARTICLE

SR-DSGA: Session Recommendation for Dual Sequence Based on Graph Neural Network and Multi-Attention

BAOJUN TIAN, NANA LIU, AND JIANDONG FANG
College of Information Engineering, Inner Mongolia University of Technology, Hohhot 010080, China
Corresponding author: Baojun Tian (ngdtbj@126.com)

This work was supported by the Natural Science Foundation of Inner Mongolia Autonomous Region, China, under Grant 2023LHMS06008.

ABSTRACT Session recommender system (SRS) captures user's sequential features based on historical behavior to predict the next-clicked item. The accuracy of extracting user's session features directly determines the key performance of SRS. Existing session recommendation methods have two flaws: 1) ignore the complex connections between items, i.e. represent them in a relatively isolated manner; 2) neglect the transition patterns between attributes of items. To address these issues, we propose a novel session recommendation model named SR-DSGA (Session Recommendation for Dual Sequence based on Graph neural network and multi-attention). Firstly, SR-DSGA adopts message passing mechanism in graph neural network to get non-isolated item embedding representations with specific semantic relationship by item-level explicit sequence modeling. Secondly, SR-DSGA exploits the Transformer's multi-head self-attention mechanism to indirectly obtain item embedding representations in another way through item attribute-level implicit sequence modeling. Therefore, SR-DSGA can help extract the fine-grained features with full sequential patterns even in sparse data scenarios. Finally, soft-attention and time threshold are used to acquire user's long-term and short-term preferences respectively. Experimental studies on real-world datasets demonstrate the proposed SR-DSGA model outperforms the state-of-the-art benchmark methods.

INDEX TERMS Session recommendation, behavior features, dual sequence modeling, graph neural network, multi-attention.

I. INTRODUCTION

With the rapid development of Web 2.0, users are not only consumers of information, but also producers and disseminators of information, leading to the rapid growth of data. In the massive data space, it is more difficult for users to obtain valuable content that meets their personalized needs. As one of the key technologies to alleviate information overload and improve user experiences, recommender system has also attracted more and more attention from academia and industry in recent years [1], [2].

As a sequence method, session recommender system exploits the behavior of anonymous users over a period of time to predict the next clicked item. It mainly includes two categories [3], [4]: (1) Conventional recommender methods include item-based neighborhood and Markov chains. (2) Deep neural network based models [5].

Although the methods above have achieved satisfactory results, they still have some limitations. Markov chain method reduces the prediction accuracy of the recommender system when only the local sequence pattern between every two adjacent items is modeled. RNN has a fatal flaw in session modeling, it ignores the connections between items and represents their embedding features in an isolated manner. Representing the session as a graph is a way to solve the above defects. Wu et al. [6] proposed a session recommendation model named SR-GNN (Session-based Recommendation with Graph Neural Network), which represents session sequences as graph structures and explores rich transitions among items and generate accurate item

The associate editor coordinating the review of this manuscript and approving it for publication was Valentina E. Balas.

embedding vectors. Liang et al. and Luo et al. [7], [8] proposed a behavior-aware session recommendation system based on graph neural network, which exploits GNN to model user's session behavior. However, the methods only consider sequential patterns between items, ignoring the key factor for capturing user's fine grain preferences, i.e. sequential patterns between item attributes. And user's behaviors involved in a session is usually sparse, so it is difficult to accurately represent the session features.

In this paper, we propose a novel session recommendation model named SR-DSGA. Firstly, the item-level sequence is modeled using graph neural networks to mitigate the problem of inaccurate embedding representations of items due to data in isolation. Secondly, the multi-head attention mechanism is used to address the problem that the graph neural network model ignore the transition patterns between item attributes, thus failing to obtain the full sequential patterns. Finally, the average time r (300 seconds) for browsing items, especially in the field of e-commerce applications, is used as a reference [9], and the time t ($\frac{r}{2} < t < r$) is set as a threshold for filtering the item sequence to obtain the user's short-term preference, and the soft attention mechanism is exploited to capture the user's long-term preference. Experimental results show that our work improves key performance indicators, such as Precision and Mean Reciprocal Rank, compared to the SOTA models, and provides a novel and feasible solution for session recommendation.

In summary, the main contributions of this work are:

(1) We propose dual sequence model to accurately capturing user's fine grain behavioral preferences even in sparse data scenarios.

(2) We distinguish between long-term and short-term preferences, further enhancing the rationality of the recommender system.

(3) To evaluate the efectiveness of SR-DSGA model, we conduct extensive experiments on real-world datasets. The experimental results show SR-DSGA's superiority over the baseline models.

The remainder of this paper is organized as follows: Section II briefly introduces related work; Section III describes the proposed model and the implementation of each part of the model in detail; Section IV elaborates experiments and evaluations; Sections V reviews the main work of the paper and discusses the future research directions.

II. RELATED WORK

A. CONVENTIONAL RECOMMENDATION METHODS

The item-based neighborhood methods are a naive solution that uses similarities between items in the same session to recommend the next item. For example, Sarwar et al. [10] proposed a item-based neighborhood method, which computes inter-item cosine similarity based on the co-occurrence of items in the same session. These approaches only exploit a last-clicked item to generate recommendation, which ignores the session order between items, resulting in inaccurate recommendations. Markov chains as a sequential optimization conventional method are employed to capture the user's behavior features to predict next action. FPMC fused matrix factorization and first-order Markov Chains to capture long-term preferences and short-term item-item transitions respectively [11]. Chen et al. [12] proposed logical Markov embeddings to capture sequence information in interactions. Wang et al. [13] proposed a hybrid representation model using a hierarchical structure to combine user and item sequence features in a nonlinear manner. These approaches predict user's next behaviors based on the previous ones, thus improving the performance of recommender systems to a certain extent. However, the assumption of independent past components of Markov chain based models confines the prediction accuracy.

B. DEEP NEURAL NETWORK BASED MODELS

Balázs et al. [14] applied the recurrent neural network (RNN) to the session-based recommender system for the first time, which mainly addressed the problem of not being able to represent the item sequence features in a session. Quadrana et al. [1] proposed a hierarchical recurrent neural network model, which can capture the user features of the current session, and can also integrate the user's historical preferences to produce better recommendations. Wu et al. [6] proposed a session recommendation model named SR-GNN. Session sequences were represented as graph structures, GNN was used to obtain item representations, and then a attention mechanism was used to capture the user's session features. Yang et al. [15] proposed the feature-level attention neural model, i.e. FANM, which used the gated recurrent unit (GRU) and multi-head attention mechanism to extract the user's current interests from the sequence of the clicked items, and then integrated the user's long-term and short-term preferences to predict the next-clicked item. It can address the problem of shifting interests and improve the accuracy of recommender system. Yao et al. [16] proposed a session recommendation model named SR-MNN, which used graph convolutional network (GCN) to extract long-term session features, and used RNN to extract short-term session features, and finally constructed a global feature of each session. Xu et al. [17] proposed a session recommendation model named SRHetGNN based on heterogeneous graph neural network. The model constructed the session sequence into a heterogeneous graph containing multiple types of nodes, then used GNN to learn the complex transitions between nodes, and finally used the attention mechanism to generate session embeddings of each node. Liu et al. [22] proposed STAMP model, which took the features of the last interaction item as the user's short-term preference, and combined the user's long-term preference. The experimental results showed the user's short-term preferences played a key role in session recommender system. Li et al. [23] proposed NARM model, which combined the recurrent neural network and the attention mechanism. The former was used to capture the user's sequence behavior features, and the latter was used to extract the user real intention.

Although the methods above achieve relatively satisfactory results, they still have some limitations. Markov chain method decreases the prediction accuracy of recommender system under strong independence assumption. RNN model fails to capture the transition relationship between distant items. It is simpler to deal with users' short-term preferences, which affects the accuracy of the session recommender system. User's behaviors involved in a session is usually sparse, these methods are difficult to accurately represent session features. And the existing methods usually only consider the transition patterns between items, but ignore the implicit transition patterns between item attributes.

In this paper, a novel dual-sequence session recommendation model, namely SR-DSGA, is proposed, that is, historical session sequences of user behaviors are reorganized as a session graph, GNN extracts the session sequence features of items, and multi-head attention mechanism captures the attribute sequence features of items.

Then the outputs of two different sequence models are aligned to obtain the user's fine-grained behavior features with full sequence patterns.

C. ATTENTION MECHANISMS

Bahdanau et al. [18] proposed attention mechanism to eliminate the defects of the encoder-decoder in the sequence model. In the encoding stage, the method is used for all hidden states, which can ensure that the last hidden layer can contain all the input information and produce different contributions to each word in the decoding stage, thus reflecting word features with different degrees of correlation between source and target sentences. Attention mechanism has three core elements, i.e. Query, Key and Value. In general, Query vector represents an element in a given target sequence, and source sequence consists of a Key matrix and a Value matrix [19].

The calculation process of attention mechanism is shown in Fig. 1. Firstly, for a given Query vector, the similarity between it and each vector of the Key matrix is calculated by formula (1), which is the weight coefficient. Secondly, the weight coefficient is normalized by formula (2). Finally, according to the weight coefficient, the vector corresponding to the Value matrix is weighted and summed, and calculated by the formula (3).

$$simi_i(Key_i^* Query) = Query^* Key_i \tag{1}$$

$$a_i = soft\max(simi_i) = e^{simi_i} \Big/ \sum_{j=1}^{n} e^{simi_j} \tag{2}$$

$$Attention = \sum_{i=1}^{n} a_i^* Value_i \tag{3}$$

This paper employs a multi-head attention mechanism to capture the session features of item attribute sequences.

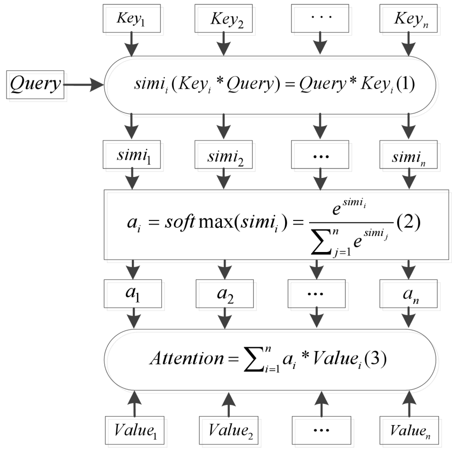

FIGURE 1. The calculation process of the attention mechanism.

Experiments show that the mechanism can indeed improve the accuracy of session recommender system.

III. THE PROPOSED MODEL

A. FRAMEWORK

The overall framework of SR-DSGA model proposed in this paper is shown in Fig. 2. The model mainly includes three parts: (1) Explicit item-level sequence modeling: each session sequence is constructed into a graph, and gated graph neural network (GGNN) is used to iteratively update the state of each node to learn the behavior representations in non-Euclidean space. (2) Implicit attribute-level sequence modeling: it can address the problems that the graph neural network model does not make full use of the sequence features of session, and item-level sequences alone cannot reveal full sequence patterns. (3) The time t ($\frac{r}{2} < t < r$, r is the average time for browsing items) is set as a threshold for filtering the item sequences to obtain the user's short-term preference. And soft attention mechanism is used to capture user's long-term preference.

B. ITEM-LEVEL SEQUENCE MODELING

Items clicked in a session are arranged in chronological order. Since a item may appear multiple times in a sequence, the sequence is expressed as a directed graph structure. The state of the node is iteratively updated by GRU to realize the transition relationship modeling between items [20].

1) CONSTRUCTING SESSION GRAPH

Let $V = \{v_1, v_2, \cdots, v_n\}$ represents a set of items that do not repeat in all sessions. Each session sequence s is expressed as $S_s = [v_{s,1}, v_{s,2}, \cdots, v_{s,m}]$, $v_{s,i} \in V$, and the items in the sequence s are arranged in chronological order. GNN first models each session sequence as a directed graph $G(v_{s,i}, e_s)$, the node in a graph correspond to the item $v_{s,i} \in V$ in a session, and directed edge $(v_{s,i-1}, v_{s,i}) \in e_s$ represents the order of clicked item. Since an item in the session sequence may appear repeatedly, a normalized weight is assigned to

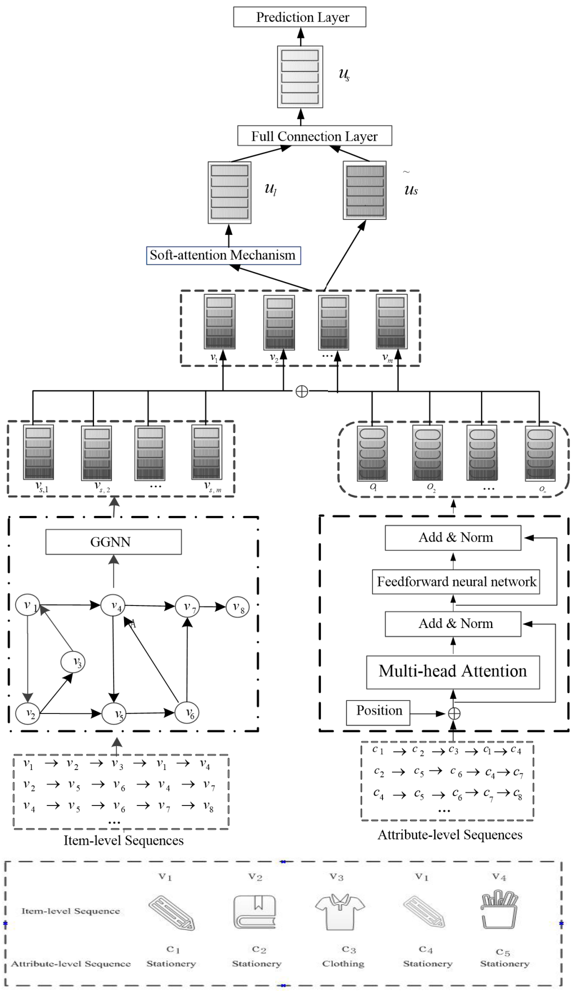

FIGURE 2. The overall framework of SR-DSGA model.

each edge, which is equal to occurrences of the edge divided by the out-degree of the start node of the edge.

2) EMBEDDING REPRESENTATION

Graph neural network is capable of mining the transition between items to generating accurate item embedding with specific semantic relationship.

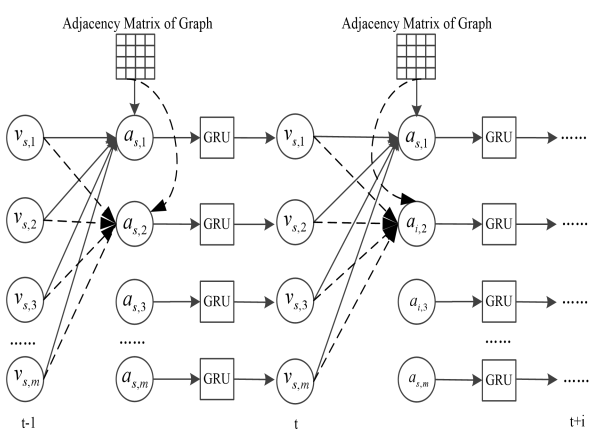

FIGURE 3. Embeding update process of graph nodes.

The adjacency matrix of a directed graph G($v_{s,i}$, e_s) is $\boldsymbol{A} \in \boldsymbol{R}^{m\times 2m}$, which is composed of in-degree matrix $\boldsymbol{A}_{In}$ and out-degree matrix $\boldsymbol{A}_{Out}$. According to the state of the current input node and the hidden state of the node at the previous moment t-1, the update gate and the reset gate of GRU model are used to update the current state of target node.

The update gate $a_{s,t}^t$ controls how much information at the current moment t and the previous moment t-1 needs to be retained. The calculation formula is as follows.

$$a_{s,i}^t = \boldsymbol{A}_{s,i}[v_1^{t-1}, v_2^{t-1} \ldots, v_m^{t-1}]^T \boldsymbol{H} + b \tag{4}$$

where $\boldsymbol{A}_{s,i} \in \boldsymbol{R}^{1\times 2m}$ corresponding to the in-degree and out-degree of each node. $[v_1^{t-1}, v_2^{t-1}, \ldots, v_m^{t-1}](v_i^{t-1} \in R^d (i = 1, 2, \ldots, m))$ is the feature matrix of m nodes at moment t-1. $\boldsymbol{H} \in \boldsymbol{R}^{d\times 2d}$ is the weight parameter.

$$z_{s,i}^t = \sigma(\boldsymbol{W}_z a_{s,i}^t + \boldsymbol{U}_z v_i^{t-1}) \tag{5}$$

where $\sigma(\cdot)$ represents the sigmoid function, $a_{s,i}^t$ is the current input, $\boldsymbol{W}_z$, $\boldsymbol{U}_z$ are the weight parameters, v_i^{t-1} is the hidden state of the node i at moment t-1.

The reset door $r_{s,t}^t$ is used to control how much information can be used from the previous moment. The calculation formula is as follows.

$$r_{s,i}^t = \sigma(\boldsymbol{W}_r a_{s,i}^t + \boldsymbol{U}_r v_i^{t-1}) \tag{6}$$

where $\boldsymbol{W}_r$, $\boldsymbol{U}_r$ are the parameters of the reset doors.

The candidate hidden state $\tilde{v}_{s,t}$ is mainly a combination of the input $a_{s,i}^t$ at the current moment t and the hidden state v_i^{t-1} at the previous moment t-1. The reset gate $r_{s,i}$ filters out the part of the hidden state at the previous moment t-1 that is independent of the current input. The calculation formula is:

$$\tilde{v}_{s,i}^t = \tanh(\overline{\boldsymbol{W}} a_{s,i}^t + \overline{\boldsymbol{U}}(r_{s,i}^t \odot v_i^{t-1})) \tag{7}$$

where $\odot$ is the Hadamard product operation between the elements of the vector. $\overline{\boldsymbol{W}}, \overline{\boldsymbol{U}}$ are parameters that control the input of the current moment and the hidden state of the previous moment t-1.

Finally, the hidden state of the input node at the current moment t is composed of the hidden state of the previous moment t-1 and the candidate hidden state. The update gate $z_{s,i}^t$ retains the valuable part of the candidate hidden state for the input of the current node. So we obtains the feature representations of the node i at moment t. The calculation formula is as follows.

$$v_{s,i}^t = (1 - z_{s,i}^t) \odot v_i^{t-1} + z_{s,i}^t \odot \tilde{v}_{s,i}^t \tag{8}$$

where $z_{s,i}^t$ is the reset door, v_i^{t-1} is the hidden state of the previous moment, and $\tilde{v}_{s,i}^t$ is the candidate hidden state.

C. ATTRIBUTE-LEVEL SEQUENCE MODELING

Existing session recommendation models only consider the transition relationship between items, but ignore the transition patterns between item attributes, which can better express user's fine-grained behavior. Explicit item-level and implicit feature-level sequences can help extract the full sequential patterns, which not only alleviates the sparsity of item-level session data by data augmentation, but also further improve the accuracy of session recommender system with complete session features.

In this paper, Transformer's encoder is exploited to extract features of attribute-level sequences [21].

1) POSITIONAL ENCODING

Since there is no positional information in the encoder structure of Transformer, positional encoding $p_{s_i} \in R^d$ is injected to the embedding vector $c_{s,i} = [c_1, c_2, \ldots, c_d]$ of the attribute of item i in the attribute-level sequence $C_s = [c_{s,1}, c_{s,2}, \ldots, c_{s,m}]$. The even dimension and the odd dimension of the p_{s_i} are calculated by employing sin and cos functions respectively.

$$p(i, 2k) = \sin(i/10000^{2k/d}) \tag{9}$$

$$p(i, 2k+1) = \cos(i/10000^{2k/d}) \tag{10}$$

where $p(i, 2k)$ and $p(i, 2k+1)$ are the 2k and 2k+1 components of the encoding vector of position i, respectively, and d is the dimension of the positional encoding vector p_{s_i}.

The attribute-level embedding is spliced with the corresponding positional vector, as shown in formula(11).

$$\boldsymbol{F}_s = \boldsymbol{C}_s + \boldsymbol{P}_s = [c_{s_1} + p_{s_1}, \ldots, c_{s_m} + p_{s_m}] \tag{11}$$

where $\boldsymbol{C}_s \in \boldsymbol{R}^{d\times m}$ is the attribute feature matrix, $\boldsymbol{P}_s \in \boldsymbol{R}^{d\times m}$ is the positional feature matrix.

2) ATTRIBUTE-LEVEL FEATURE REPRESENTATIONS

According to section II-C, the attention calculation formula is as follows.

$$\mathrm{SDPA}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \mathrm{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}})\boldsymbol{V} \tag{12}$$

where $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ represent query matrix, key matrix, and value matrix respectively,d is feature vector dimension.

In order to enhance the attribute feature expression ability in different feature subspaces, this paper uses a multi-head self-attention. $\boldsymbol{F}_s$ is converted to three different matrices through linear transformation, and then feed it into the SDPA. The calculation formula is as follows.

$$\boldsymbol{H}_i = SDPA(\boldsymbol{F}_s\boldsymbol{W}_{\mathrm{i}}^Q, \boldsymbol{F}_s\boldsymbol{W}_i^K, \boldsymbol{F}_s\boldsymbol{W}_i^V) \tag{13}$$

where $\boldsymbol{W}_i^Q, \boldsymbol{W}_i^K, \boldsymbol{W}_i^V$ are the projection matrices.

The multi-headed attention is defined as follows.

$$\boldsymbol{H}_s = \mathrm{Contact}(\boldsymbol{H}_1, \boldsymbol{H}_2, \cdots, \boldsymbol{H}_{hd})\boldsymbol{W}_h \tag{14}$$

where hd is the number of heads and $\boldsymbol{W}_h$ is the model parameter.

And the model employs a residual module, a layer normalization and fully connected layer with a ReLU activation function to avoid the problems of gradient disappearance and networks devolution in the training process. The process is as follows.

$$\boldsymbol{O}_1 = \mathrm{LayerNorm}(\boldsymbol{F}_s + \mathrm{Muti}(\boldsymbol{F}_s)) \tag{15}$$

$$\boldsymbol{O}_s = \mathrm{LayerNorm}(\boldsymbol{O}_1 + \mathrm{F_f}(\boldsymbol{O}_1)) \tag{16}$$

where Muti($\boldsymbol{F}_s$), that is H_s, is the output calculated by the multi-head attention mechanism. $F_f(\boldsymbol{O}_1)$ represents the output of the feedforward neural network, and its calculation formula is as follows.

$$\mathrm{F_f}(\boldsymbol{O}_1) = Relu(\boldsymbol{W}_1 * \boldsymbol{O}_1 + b_1)\boldsymbol{W}_2 + b_2) \tag{17}$$

where $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, b_1, b_2 are model parameters. $O_s \in \boldsymbol{R}^{m\times d}$ is the feature matrix learned by Encoder in the Transformer model.

3) USER'S SHORT-TERM AND LONG-TERM PREFERENCES

Item-level feature and attribute-level feature are aligned and fused to obtain full sequential representation named U_s.

$$\boldsymbol{U}_s = \boldsymbol{W}_s[\boldsymbol{S}_s \oplus \boldsymbol{O}_s] \tag{18}$$

where $\mathbf{U}_s = (v_1, v_2, \ldots, v_m)$, $v_i \in R^r$, $\boldsymbol{U}_s \in \boldsymbol{R}^{r\times m}$, $\boldsymbol{S}_s = (v_{s,1}, v_{s,2}, \ldots, v_{s,m})$, $v_{s,i} \in R^d (i = 1, 2, \ldots .m)$ are the item-level feature matrix. $\mathbf{O}_s = (O_1, O_2, \ldots, O_m)$, $O_i \in R^d$ (i =1,2,...,m) is the attribute-level feature matrix. $\boldsymbol{W}_s \in \boldsymbol{R}^{r\times d}$ is the parameter matrix.

Since user's preferences are dynamic, it is necessary to distinguish between long-term and short-term preferences. Firstly, to reduce the impact of noisy data on recommender system, we take the time t ($\frac{\mathrm{r}}{2} < \mathrm{t} < \mathrm{r}$, r is the average time for browsing items) as a threshold for filtering the item sequences to obtain the user's short-term preference. Suppose that A is the last item that is browsed more than t, then the feature of item A is taken as the short-term preference, i.e. $\tilde{u_s}$. Secondly, the soft attention mechanism is used to capture the long-term preference. The formula is as follows.

$$a_i = z^T\sigma(\boldsymbol{W}_k v_k + \boldsymbol{W}_l v_i + b) \tag{19}$$

where $z \in R^r$, $\boldsymbol{W}_k \in \boldsymbol{R}^{r\times r}$, $\boldsymbol{W}_l \in \boldsymbol{R}^{r\times r}$, $b \in R^r$ are the parameters of the model.

Then, user's long-term preference u_l is as follows.

$$u_l = \sum_{i=1}^{m} a_i * v_i \tag{20}$$

Finally, the user's preference is obtained as follows.

$$u_s = \boldsymbol{W}_s[\tilde{u}_s; u_l] \tag{21}$$

where $\boldsymbol{W}_s \in \boldsymbol{R}^{r\times 2r}$ is the parameter matrix.

D. MODEL TRAINING

The recommendation score r_i is computed by multiplying user's preference u_s by each candidate item $v_i \in V$ as follows.

$$\mathrm{r_i} = u_s^T v_i \tag{22}$$

Then, the recommended scores r input softmax function is converted into probability, and the predicted value z' is obtained as follows.

$$z' = \mathrm{soft\,max}(r),\ r = [\mathrm{r_1,r_2,\ldots,r_n}] \tag{23}$$

The model is trained by the cross entropy loss function, which is shown in formula(24) as follows.

$$l(z') = -\sum_{i=1}^{n} z_i \log(z'_i) + (1 - z_i)\log(1 - z'_i) \tag{24}$$

where z_i is the true label value.

IV. EXPERIMENTS

A. ENVIRONMENT AND DATASETS

We use GPU Nvidia Tesla P100 server as the computing platform, Ubuntu16.4 as the operating system, pycharm 2021.3 as development tool, python 3.7.0 as development language, and pytorch 1.4 as deep learning framework.

The real data set is Diginetica, which is provided by CIKM Cup 2016 [6]. Its content includes records of user interactions with items, and attribute information for items.

We use 5-fold cross-validation to obtain reliable experimental results. Specifcally, we randomly divide all sessions in the dataset into five equal parts. We select one of the five parts as the test set and the remaining as the training set for each experiment. Besides, we randomly select half of the sessions in the test set to form the validation set.

B. METRICS

In this paper, Precision and Mean Reciprocal Rank (MRR) are selected as the metrics. Precision is used to measure the prediction accuracy of model. The calculation method is as follows.

$$\mathrm{P@N} = \frac{\mathrm{n_{hit}}}{\mathrm{n}} \tag{25}$$

where $\mathrm{n_{hit}}$ represents the number of correct samples in the first N items in the recommendation list, and n represents the total number of samples in the test set.

MRR@N is used to measure the rationality of the ranking and represent the mean reciprocal rank of the correctly recommended items. Its calculation formula is as follows.

$$\mathrm{MRR@N} = \frac{1}{\mathrm{n}} \sum_{\mathrm{i \in M}} \frac{1}{\mathrm{rank_i}} \tag{26}$$

where n refers to the number of samples in the test set, M refers to the number of samples containing the correct recommended items in the first N recommended items, and $\mathrm{rank_i}$ refers to the ranking of item i in the recommended lists.

C. HYPERPARAMETERS TUNING

To obtain the best performance for SR-DSGA model, we finetune the most important hyperparameters on the validation set.

1) NUMBER OF HEADS

Fig. 4 shows the effect of the number of heads in the attention mechanism on the performance indicators MRR@20 and P@20. It can be seen that with the number of heads increasing, the values of MRR@20 and P@20 are higher and then lower gradually. When the number of heads is set to 4, the two values reach the maximum, and the performance of the model is optimal.

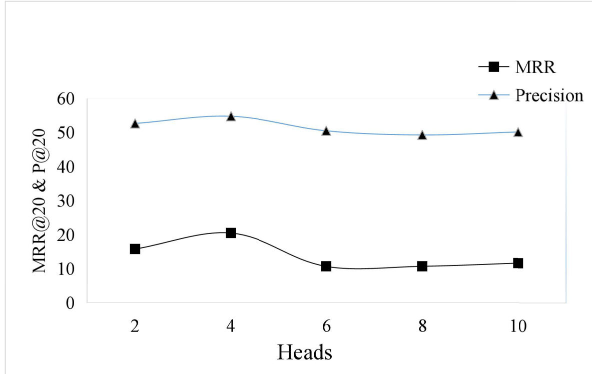

FIGURE 4. The impact of heads on MRR and precision.

2) EMBEDDING DIMENSIONS

The initial embeddings of items and attributes as low-dimensional dense vectors, and feeds the vectors into graph neural networks and multi-head attention mechanisms to learn their feature representations. From the experimental results in Fig. 5, it can be seen that when the dimension is smaller (e.g. 40), the learning ability of the model is lower, and when the dimension is larger (e.g. 120), the model appear overfitting. When the dimension is equal to 100, the key performance of SR-DSGA model is optimal.

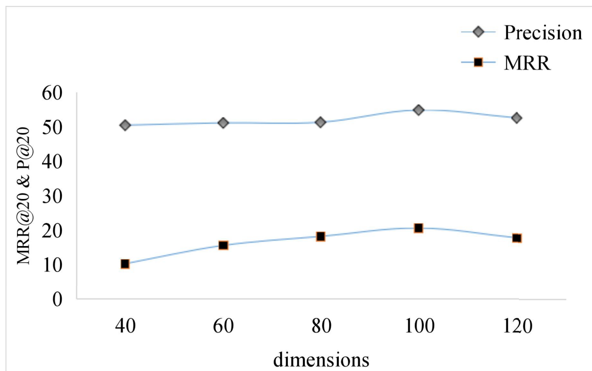

FIGURE 5. The impact of embedding dimension on MRR and precision.

D. BENCHMARKS AND RESULTS ANALYSIS

To evaluate the performance of SR-DSGA model, we compare it with the following benchmarks:

SR-GNN [6] uses graph neural networks to extract user's behavior features in session sequences.

STAMP [22] takes the features of the last interaction item as the user's short-term preference, and combines the user's long-term preference. The model has achieved better results, and also shows that the user's short-term preference plays a key role in session recommendation system.

NARM [23] combines RNN and attention mechanism. The former is used to capture the user's sequence behavior features, and the latter is used to extract the user's real intention.

TABLE 1. Comparison of different models on MRR and precision.

Models	MRR@10	MRR@20	P@10	P@20
NARM	15.01	16.17	36.72	49.70
STAMP	16.05	17.01	37.05	45.64
SR-GNN	17.04	17.59	38.40	50.73
SR-DSGA	**17.74**	**20.48**	**41.63**	**54.76**

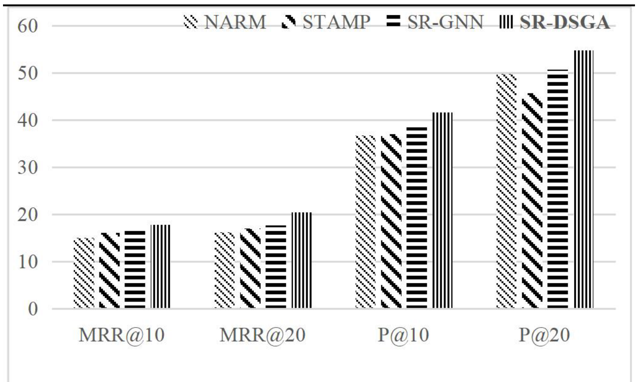

FIGURE 6. Experimental results of SR-DSGA and benchmarks.

Tab. 2 and Fig. 6 present the recommendation perfor-mance of all compared models on Diginetica dataset. The proposed SR-DSGA performs better in Precision and MRR than current SOTA models, such as SRGNN, STAMP and FDSA, on the same dataset, respectively. Experimental results indicate that SR-DSGA can help to extract the fine-grained session features with full sequential patterns, i.e. item-level explicit sequences and attribute-level implicit sequences, to improve the critical performance of session recommender system.

E. ABLATION ANALYSIS

In this subsection, to evaluate the effectiveness of each key component, we design the simplifed variant of SR-DSGA, namely SR-GNN and SR-ATT, with only item-level sequences and attribute-level sequences, respectively.

TABLE 2. Ablation analysis on key component of SR-DSGA.

Models	MRR@10	MRR@20	P@10	P@20
SR-ATT	16.59	17.31	37.77	46.16
SR-GNN	17.04	17.59	38.40	50.73
SR-DSGA	**17.74**	**20.48**	**41.63**	**54.76**

Tab.2 list the results of ablation analysis, showing the values of MRR@10 and MRR@20 in the proposed SR-DSGA are higher than the corresponding values of SR-GNN and SR-ATT by 0.7 and 2.89 and 1.15, 3.17, respectively. Moreover, the values of P@10 and P@20 in the proposed SR-DSGA are higher than the corresponding values of SR-GNN and SR-ATT by 3.23, 4.03 and 3.86, 8.6, respectively. Ablation analysis further elucidates that the proposed dual sequence model based on graph neural network and multi-attention is feasible and effective for improving the key performance of session recommender system.

V. CONCLUSION

This paper proposes a novel session recommendation model named SR-DSGA. The model organizes the session data into a directed graph structure, and uses the message propagation method of graph neural network to aggregate the nodes, so as to realize the explicit sequence modeling of item transition patterns. At the same time, the multi-head attention mechanism of Transformer is used to realize the implicit sequence modeling of distance-independent item features, which not only compensates for the fact that the graph neural networks approach does not take full advantage of orderliness of the data, but also but also alleviates the influence of data sparsity on model performance through data augmentation. Finally, the time t ($\frac{r}{2} < t < r$, r is the average time for browsing items) is set as a threshold for filtering the item sequences to obtain the user's short-term preference and soft attention mechanism is used to capture user's long-term preference. We have conducted extensive experiments on real datasets, and the results show that the SR-DSGA proposed in this paper outperforms the state-of-the-art benchmark methods. In addition to accuracy, explanation, diversity, and robustness are also important future research directions in the field of session recommender systems.

REFERENCES

[1] M. Quadrana, A. Karatzoglou, B. Hidasi, and P. Cremonesi, "Personalizing session-based recommendations with hierarchical recurrent neural networks," in *Proc. 11th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 130–137.

[2] D. D. Woods, E. S. Patterson, and E. M. Roth, "Can we ever escape from data overload? A cognitive systems diagnosis," *Cognition, Technol. Work*, vol. 4, no. 1, pp. 22–36, Apr. 2002.

[3] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.

[4] S. J. Wang, L. B. Cao, and Y. Wang, "A survey on session-based recommender systems," *ACM Comput. Surveys (CSUR)*, vol. 54, pp. 1–38, Jul. 2022.

[5] C. Wang, W. Ma, C. Chen, M. Zhang, Y. Liu, and S. Ma, "Sequential recommendation with multiple contrast signals," *ACM Trans. Inf. Syst.*, vol. 41, no. 1, pp. 1–27, Jan. 2023.

[6] S. Wu, Y. Tang, and Y. Zhu, "Session-based recommendation with graph neural networks," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2019, pp. 346–353.

[7] Y. Liang, Q. Song, Z. Zhao, H. Zhou, and M. Gong, "BA-GNN: Behavior-aware graph neural network for session-based recommendation," *Frontiers Comput. Sci.*, vol. 17, no. 6, pp. 109–132, Dec. 2023.

[8] J. Luo, M. He, W. Pan, and Z. Ming, "BGNN: Behavior-aware graph neural network for heterogeneous session-based recommendation," *Frontiers Comput. Sci.*, vol. 17, no. 5, Oct. 2023, Art. no. 175336.

[9] Y. Sun, "Research and implementation of personalized recommender system based on deep learning," M.S. thesis, Beijing Univ. Posts Telecommun., Beijing, China, 2019.

[10] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 1–23.

[11] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World wide web*, Apr. 2010, pp. 811–820.

[12] S. Chen, J. L. Moore, D. Turnbull, and T. Joachims, ''Playlist prediction via metric embedding,'' in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2012, pp. 714–722.
[13] P. F. Wang, J. F. Guo, and Y. Y. Lan, ''Learning hierarchical representations model for next basket recommendation,'' in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2015, pp. 403–412.
[14] H. Balazs, K. Alexandros, and B. Linas, ''Session-based recommendations with recurrent neural networks,'' 2016, *arXiv:1511.06939*.
[15] Q. Yang, P. Luo, X. Cheng, N. Li, and J. Zhang, ''Feature-level attentive neural model for session-based recommendation,'' *IEEE Access*, vol. 8, pp. 132582–132591, 2020.
[16] H. Yao, J. Hu, W. Xie, Y. Huang, and W. Xie, ''Session-based recommendation model based on multiple neural networks hybrid extraction feature,'' in *Proc. IEEE Int. Conf. Big Data*, Dec. 2020, pp. 5315–5322.
[17] L. Xu, W.-D. Xi, and C.-D. Wang, ''Session-based recommendation with heterogeneous graph neural networks,'' in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.
[18] D. Bahdanau, K. Cho, and Y. Bengio, ''Neural machine translation by jointly learning to align and translate,'' in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–15.
[19] A. Aljohani, M. A. Rakrouki, N. Alharbe, and R. Alluhaibi, ''A self-attention mask learning-based recommendation system,'' *IEEE Access*, vol. 10, pp. 93017–93028, 2022.
[20] W. Chen, M. K. He, Y. X. Ni, and W. K. Pan, ''Global and personalized graphs for heterogeneous sequential recommendation by learning behavior transfers and user intention,'' in *Proc. 16th ACM Conf. Recommender Syst.*, 2022, pp. 268–277.
[21] E. Yuan, W. Guo, Z. He, H. Guo, C. Liu, and R. Tang, ''Multi-behavior sequential transformer recommender,'' in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 1642–1652.
[22] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, ''STAMP: Short-term attention/memory priority model for session-based recommendation,'' in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 1831–1839.
[23] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, ''Neural attentive session-based recommendation,'' in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 1419–1428.

BAOJUN TIAN received the M.S. degree from Inner Mongolia University of Technology, Hohhot, China, in 2003. He is currently a Professor. His research interests include recommender systems, machine learning, and natural language processing.

NANA LIU received the M.S. degree from Inner Mongolia University of Technology, Hohhot, China, in 2022. Her research interest includes recommender systems.

JIANDONG FANG received the M.S. degree from Taiyuan University of Technology, in 1998, and the Ph.D. degree from Tianjin University, China, in 2009. She is currently a Professor. Her research interests include intelligent control and intelligent system theory and its application, and embedded intelligent control and intelligent systems.