IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 26, 2019, accepted August 15, 2019, date of publication August 20, 2019, date of current version August 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2936461

GACOforRec: Session-Based Graph Convolutional Neural Networks Recommendation Model

MINGGE ZHANG AND ZHENYU YANG

School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China

Corresponding author: Zhenyu Yang (yang_zhenyu@163.com)

This work was supported in part by the Shandong Natural Science Fund Project under Grant ZR2017LF021.

ABSTRACT The biggest challenge to recommendation systems based on user preferences is how to improve the ability of the recommendation system to mine and analyse user preferences and behaviours. In this process, we must not only consider the continuation of the user's long-term preference but also improve the system's ability to accommodate short-term preferences and discrete preferences. To this end, we focus on the performance of time factors of user preferences. However, the issue we are concerned about has not received much attention in the existing research. We propose a new recommendation model based on the perspective of user sessions, namely GACOforRec. This model can handle long-term and stable preferences at the same time and preserve the hierarchy of potential preferences. We conducted a large number of comparative experiments on two real datasets, and the results show that GACOforRec is significantly better than other state-of-the-art methods in the study of user sessions.

INDEX TERMS Session, recommendation system, graph convolutional networks, spatial-temporal information, attention mechanism.

I. INTRODUCTION

The recommendation system is the product of this information overload era. It can recommend items to users based on their interest in massive network data. At present, the recommender systems have been applied in various domains such as music recommendation, movie recommendation, book recommendation, product recommendation [4]. Such a function not only can recommend the products that the users may like but also can actively promote the marketing of the products.

Of course, the foundation of the recommendation system is built on the assumption that the Internet history of users can completely represent the preferences of users [15], and this assumption means that user preference can be obtained by data mining and analysis on user history [16], [24]. However, user preferences and behaviours are not always stable for long periods. User preferences can be roughly divided into three situations: long-term stable preferences, discrete and fluctuating preferences, and short-term preferences [29]. We have to consider as many cases as possible to ensure that the system can adapt to changes in user preferences [19]. Based on this consideration, we believe that there should be purposeful modelling of user behaviour in the session. Besides, it is more meaningful to consider the message sharing between different sessions to generate the corresponding recommendations.

There is a lot of data in the recommender system that does not have a regular spatial structure. For the complicated relationship between users and items, we believe that we should look for a network structure that can process both temporal and spatial information [14], [20]. The Graph Convolutional Networks (GCNs) can perform deep learning of Spatio-Temporal information on graph data, and the ability to spread objects in the graph will be significantly enhanced. The Graph Convolutional Networks has this capability, which enables deep learning of Spatio-Temporal information on graph data, and dramatically improves the ability of objects to spread within the graph.

Since the low order approximation of GCNs reflects the character of the short-term preference process. We applied ConvLSTM [21] to our method to ensure that our model can take into account more situations. This model allows the network to mine not only the temporal and spatial information learned by the GCNs but also the LSTM's ability to update and remember long-term preferences. At the same time, we propose a new adaptive attention mechanism based on GCNs to pay attention to the influence of different propagation distances of GCNs. To enhance the model's hierarchical learning of various preferences, we introduced

The associate editor coordinating the review of this article and approving it for publication was Nuno Garcia.

ON-LSTM, a network structure that focuses more on hierarchy and neuron ordering. This ordering is essential to the overall perception of the model's user preferences.

In general, in this work, we propose a session-based recommendation system with different temporal attention mechanisms based on graph convolution operations. The algorithm uses the GCNs as the model basis to abstract the actual role of user preferences in the whole application scenario and applies ConvLSTM and ON-LSTM to expand the algorithm's ability to mine different temporal and different levels of the entire behaviour layer. Our algorithm has been validated on two large datasets. The experimental results show that the algorithm performs better than the excellent current algorithms.

II. RELATED WORK

The recommendation method based on session has only become popular in recent years [4], [18], [28]. To conclude, the current mainstream methods can be divided into two different branches: Recurrent Neural Networks (RNNs) based methods [25], [31], [32] and Markov chains based approaches [7], [9]. The purpose of both is to explore the user preference timing relationship in the sessions. In 2017, Tuan et al. combined RNNs with session recommendations to mine information from time changes in user behaviour and implement recommendations [12]. Hidasi *et al.* [10] studied the application of the RNN and session in the next item recommendation, which used Gated Recurrent Neural Networks to redefine the classic RNNs. Reference [7] focuses on the dynamics of user preferences from the perspective of Hidden Markov Models (HMMs). However, this algorithm is very computationally intensive because it calculates the probability that each item is transferred to another item and continues to iteratively.

Meanwhile, [11] proposed a new method, the heuristic-based session K-Nearest Neighbor (KNN) scheme. This method outperforms loop-based network recommendations in most test configurations and the datasets at that time. This cluster-based method is essentially a feature analysis of non-matrix structures, which is more in line with the traditional RNN-oriented method of standard fabric. The fantastic performance of this method is compelling proof of the strong local correlation of actions in the session class problem, and subsequent work should be carried out based on this relationship.

In addition to the idea of clustering, researchers have considered the same problem from a different perspective. Reference [17] used two multi-layer perceptrons (MLP) to construct a model (STAMP) that can learn user intent from current behaviour. The first MLP is used to extract the user's general interests from the historical click information, and the second MLP is built on the embedding vector of the last click of each session to represent the user's current interest. However, the model only retains the timing of the internal session, but still does not give more consideration to space. This process is not enough.

It shall be noted that, as mentioned in [27], there are still some shortcomings in such methods. For example, when using RNNs, the number of users in the session is limited. When using the Markov model, only two adjacent items are used. The one-way transfer relationship is modelled while ignoring other items in the session. Some researchers have now recognized the importance of considering both temporal and spatial information [3], [8], [22], [26]. Reference [35] proposed a model that pays more attention to time factors (Time-LSTM) and simulates the time interval recommended by the next project by equipping the LSTM with a time gate. But just focusing on the effects of time factors is not enough. For the first time in 2018, [33] applied the Spatio-temporal interval to the recommendation of the next item and ST-LSTM was proposed. The model adds distance gates based on the time gate to model the user's long-term and short-term preferences. Reference [5] proposed the Variational Latent Variable Model with Recurrent Temporal Dependencies for Session-Based Recommendation (VLaReT). They focus on periodic time dependence by combining Recurrent Neural Networks with Amortized Variational Inference (AVI) to improve the predictive power of user behaviour sequences. However, these tasks are all centred on-time information, and spatial information has not received much attention.

The result is the demand for a new structure for high-level integrated timing information. Combining with the Graph Neural Networks (GNNs), a practical model solving irregular data format problem, a new session-based recommendation model is proposed: SR-GNN [27]. In SR-GNN, the author used a Gated Graph Neural Networks [1], [6], the GRU unit, to learn the hidden sequential state of users within the session, and achieved quite good results. As a variant of GNNs, the Graph Convolutional Networks (GCNs) [13] proposed by Kipf and Welling can mine more in-depth information in graphs with less computation. In our work, GCNs are applied with a pair of attention mechanisms proposed by us to learn the user's performances in different scenarios.

III. OUR PROPOSED MODEL: GACOforRec

In this section, we introduce our proposed model, GACOforRec. We present an overview of the model and a network structure diagram of the model and then introduce the details of the model in the second section.

A. OVERVIEW

We first consider the importance of user sessions. In actual application scenarios, long-term historical records may not be vital to the user. That means we should also consider the user's orderly actions within a session. Therefore, considering the order within the user's every session and the relevance between multiple sessions becomes an important task goal.

We use GCNs to model user sessions to learn the order within the session and the spatiality within the network to capture the short-term preferences of users. To avoid ignoring the user's long-term, stable preferences, we introduced

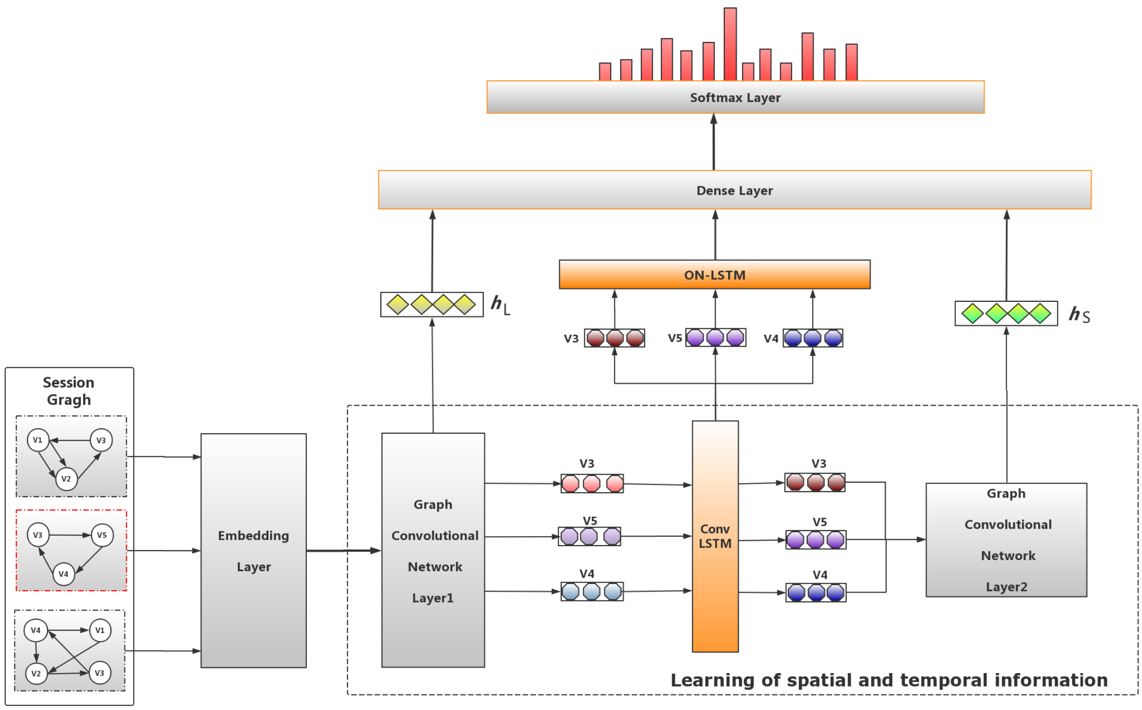

FIGURE 1. Our Model: GACOforRec. The model is roughly divided into three parts. The bottom left is the construction process of the session diagram, the bottom right is the learning process of the internal time and space information of the model, and the top is the final ranking and recommendation results of the model.

a variant of the long-and short-term memory network: ConvLSTM. The network has both spatial information processing capabilities of convolutional neural networks and the ability of Recurrent Neural Networks to preserve and update sequence information (especially long-term information). Considering that users' different behaviours may have varying degrees of impact, we propose a new pair of attention mechanisms, namely Long-Attention and Short-Attention. They can use the different propagation distances in the graph convolutional network to obtain weights [23], [34].

Besides, after adding attention to the model, we found that there is a hierarchical effect in the correlation before the different behaviours of the users learned by the model. To this end, we have hierarchically combed the disordered neurons through another variant of LSTM, ON-LSTM. To facilitate understanding of the details of subsequent models, we present the overall framework of the model here. From Fig. 1, we can visually see the network structure of the model.

B. MODEL DETAILS

According to the basic structure of the model, we introduced in III-A, we propose a session recommendation system with different temporal attention mechanisms based on graph convolution operations. The basis of the whole algorithm is based on the GCNs layers. To enhance the different levels of attention to the information in different temporal states, we use ConvLSTM and ON-LSTM separately from the GCNs to describe different impact time and different levels of influence in our algorithm. Next, we will analyse each part in detail.

1) CONSTRUCTION OF SESSION GRAPH

In the construction of the session graph $\mathcal{G}_\mathcal{S} = (\mathcal{V}_\mathcal{S}, \mathcal{E}_\mathcal{S})$, we adopt a method similar to that in SR-GNN, that is, the graph structure obtained by the user's click sequence. In the session diagram, each node represents an item $v_{s,i-1} \in \mathcal{V}$, and the user clicks on the item in the session as the edge of the graph, denoted as $(v_{s,i-1}, v_{s,i}) \in \mathcal{E}_\mathcal{S}$. At the same time, we also assign a normalised weight to each edge, that is, calculate the number of occurrences of the edge divided by the degree of the starting node of the edge, to deal with the occurrence of duplicates in the sequence. We embed each item in a unified embedding space. Node vector $v \in \mathbb{R}^d$, representing the potential vector of the item learned through the Graph Neural Network, where d is the dimension. Thus, each session can be represented as an embedded vector consisting of the node vectors used in the session graph.

2) LEARNING OF GRAPH CONVOLUTIONAL NEURAL NETWORK

In our model,session graph $\mathcal{G}_\mathcal{S} = (\mathcal{V}_\mathcal{S}, \mathcal{E}_\mathcal{S})$ after embedding the layer, the nodes v are mapped to $\mathcal{X}$, and $\mathcal{E}$ is calculated to get $\mathcal{A}$. At this point the graph can be expressed as $\mathcal{G} = (\mathcal{V}, \mathcal{X}, \mathcal{A})$, where $\mathcal{V}$ represents the set of all nodes, $\mathcal{X} \in \mathbf{R}^{N\times D}$ is composed of the feature of each node, D is the number of feature channels, and N is the number of nodes. $\mathcal{A}$ is the adjacency matrix of the graph and is used to indicate whether there are edges between the nodes. If there is an edge from v_i to v_j, then $\mathcal{A}_{ij} = 1$, otherwise $\mathcal{A}_{ij} = 0$. Each edge in the embedding graph can be represented as $e = (v, v') \in \mathcal{V} \times \mathcal{V}$, which is assigned with different feature values. The working mechanism of the Graph Neural Network is the process of continuously spreading the state information of each point in the graph to its neighbours and finally converging. For some graph with a large amount of nodes and features, such as session graph in this work, the amount of computation for the information dissemination process will be enormous. Here a Chebyshev polynomial method is implemented to reduce algorithmic computational complexity. By derivation, we can get the special graphic convolution operation as a series of multiplication:

$$h_\theta \circ x = U h_\theta U^T x \tag{1}$$

where $h_\theta = diag(\theta)$ denotes a parameterized Fourier filter. The matrix U is the eigenvector matrix of normalised Laplacian matrix, where $L = I_N - D^{-1/2} A D^{-1/2}$, which contains the structure information of the graph. To greatly simplify this complex multiplication calculation, based on the method proposed in [13], we get a multi-order approximation to $h_\theta(\Lambda)$:

$$h_\theta \circ x \approx \sum_{k=0}^{\kappa} \theta_k T_k(\tilde{L}) x \tag{2}$$

where $\tilde{L} = \frac{2}{\lambda_{max}} L - L_N$. Note that he approximate order κ here represents κ-order local information, and its physical meaning represents the distance that each node's state can reach.

3) GCNS IN GACOforRec

According to the analysis above, the complicated calculation process of the propagation process of the information in the graph can perform the κ-order approximation by the polynomial method determined by the parameter κ depending on the specific graph structure. The order reached by the polynomial approximation corresponds to the range in which each node information in the graph structure plays a role in the propagation process. Such a process coincides with the problem of the recommendation system we are concerned with. In such a process, the information on the time axis is implied. Therefore, we take a multi-layer 1^{st}-order approximation of the GCN layer. This process corresponds to a preference transfer process for users in a short time interval. The process of preference transfer of users within the segment within the entire item space. The multi-layer structure we use can gradually expand the number of features used to describe the object space, enhancing the ability to describe the feature space. The increase of the feature dimension can facilitate the classification ability of the subsequent classification network, and can also enhance the generalisation ability to a greater extent. As shown in Fig. 2, it is the basic structure of a single short-term preference transition part that we proposed. In the figure, we can see that the preference information of each

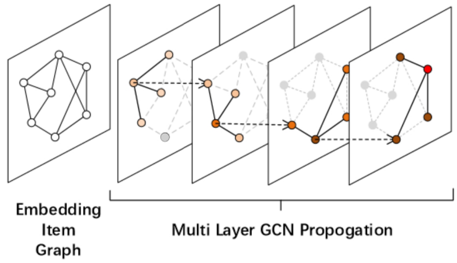

FIGURE 2. The propagation process of GCNs. We present the multi-layer GCNs propagation process to see the status of the item in the graph visually.

node can be propagated to the distance of the courtyard within a short period.

4) LONG-SHORT ATTENTION MECHANISM

To enhance the calculation of the node weights of the graphs in the model for different distances, we propose a new attention mechanism, namely the Long-Short Attention Mechanism. As described in [27], plenty of experiments and experience have shown that it is essential to put different attention weights on various parts in the recommendation process. In our work, we pay more attention to the items that are closest to the user because they have the closest relationship with the current node. We use Short Attention to represent the weight of the item with a distance of 1 from the current node:

$$h_S = \sum_{k=1}^{n} v^{(1)} \tag{3}$$

Among them, we use $v^{(1)}$ to represent the set of all item nodes with a distance of 1 from the current node. Note that since the GCNs with a layer number of one is limited by the receptive domain, in our model, the Graph Convolutional Networks of one layer is only used to extract the Short-Attention here. Relatively, Long-Attention is the attention vector of the graph convolutional layer from $k = 5$, which we use to represent the weight of each item with a distance of 5 from the current item:

$$\alpha_i = g^T \sigma(W_1 v_n + W_2 v_i + c) \tag{4}$$

$$h_L = \sum_{k=1}^{n} \alpha_i v_i \tag{5}$$

Finally, we merge the two parts of the attention vector to get a hybrid embedding:

$$h_{L,S} = W_3 \{h_L : h_S\} \tag{6}$$

So far, we have achieved the goal of simultaneously paying attention to the user's short-term preferences and discrete preferences that are slightly longer than the current click time. Experiment results show the practical value of our attention mechanisms and the selection of parameters.

5) CONVLSTM: INTRODUCED TO ENHANCE SPATIAL INFORMATION EXTRACTION ABILITY

In the preceding, a reasonable abstraction of the cross-information domain problem involving the use of GCNs for the user's preference process including time-space information is mentioned, but in actual experiments, it is found that if the number of layers and the number of states of the Graph Convolutional Networks used are insufficient. It does not exactly match the data application scenario we are currently focusing on. After analysis, the reasons are: first, the current weak graph convolution cannot fully explain the whole model. At present, the structure we have proposed is only enough to deal with the transfer process in a short period. At present, there is still a lack of consideration for those actions with long-term effects, so new timing structures are needed specifically to address long-term preferences. Second, the feature space that can be achieved by the current graph convolution is not large enough. It should also enhance the description of the problem characteristics of the network. Due to the large scope of the object space that needs to be faced in the recommendation system, we need a larger generalisation ability and object space description ability. Taken together, we got ideas from RNNs' variant LSTM, using ConvLSTM proposed by Shi, Xingjian et al. in 2015. This is a kind of Recurrent Neural Network with long and short time effects and at the same time enabling the algorithm to focus on spatial domain information. We use this structure to connect a two-segment Graph Convolutional Networks to simulate the effects of "long" and "short" effects throughout the application scenario. We use ConvLSTM to combine connections and use it to extract temporal information while paying attention to spatial feature extraction capabilities. The core nature of ConvLSTM is still consistent with LSTM, which takes the output of the previous layer as the input for the next layer.

The difference is that after the convolution operation, not only the timing relationship can be obtained, but also the spatial features can be extracted like a convolutional layer. It is worth mentioning that ConvLSTM predicts the characteristics of the central grid by the characteristics of the points around the network. Among them, the switching between state and state is calculated by a convolution operation. The specific network learning structure is as follows: ⊗ indicates a convolution operation.

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}) \tag{7}$$

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{8}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \tag{9}$$

$$\hat{c_t} = f_t \otimes c_{t-1} + i_t \otimes tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{10}$$

$$h_t = o_t \otimes tanh(c_t) \tag{11}$$

As can be seen from the above formula, ConvLTSM pays attention to the extraction of spatial features in its working mechanism, which greatly enhances the spatial searchability and problem generalization ability of our existing structures. In the actual model, we use this structure of ConvLSTM, mainly based on (9), to extract the local information of the structure around each item in the figure. Arranging such a long and short memory structure in two 5-layer GCNs directly is to better understand the two concepts of "long" and "short". ConvLSTM is a processing module for users' long-term personal preference. A different 5-layer GCNs structure corresponds to the internal relationship of different periods, and this method also shows a certain degree of attention related to some extent.

6) ON-LSTM: INTRODUCED TO HIERARCHICAL USER PREFERENCES

In the previous discussion, the necessary process of the application scenario has been modelled and simulated, but in general, this process is also part of the middle and lower layers in the entire application scenario. Therefore, we need a multi-level hierarchical structure to give equal attention to different levels in this issue.

In January 2019, Yikang Shen et al. proposed the ON-LSTM structure. ON-LSTM can distinguish between high and low level information, sorting out unordered neurons in a specific order, and then recombining them in a hierarchical structure. High-level information means that it stays longer in the higher-level interval, while the lower-level one means it is more easily forgotten in the corresponding interval. This allows the user's long-term preference information to be better preserved. The network structure can be represented by the following formula:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{12}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{13}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{14}$$

$$\hat{c_t} = tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{15}$$

$$h_t = o_t \otimes tanh(c_t) \tag{16}$$

Among them, ⊗ indicates that the corresponding elements of the matrix are multiplied. Note that the current input x_t level is higher than the history in the history before each C_t update. The way to update C_t is as follows:

$$C_t = \left\{ \begin{array}{c} \hat{c_t} < d_f \\ f_{t,[d_f:d_i]} \otimes C_{t-1,[d_f:d_i]} + i_{t,[d_f:d_t]} \otimes \hat{C}_{t,[d_f:d_i]} \\ C_{t-1} > d_i \end{array} \right\} \tag{17}$$

where d_f and d_i are hierarchical information indicating the history information h_{t-1} and the current input x_t. Note that $d_f > d_i$, which means that the history and the current input x_t do not intersect each other, then the part for $[d_i : d_f]$ is blank, and h_{t-1} is set to 0 if there is no need to update C_t.

That is to say, if the information input at the current moment has a convergence, the information at a low level is integrated into the information of the high level. We believe that in this layer, we can let the network learn the hierarchical structure information unsupervised. The hierarchical structure leads to different spans of different models, that is, the transmission distance of varying levels of information is different, which will eventually lead us to the hierarchical structure. Matrix representation is made to integrate into the neural network.

The addition of such a configuration in our algorithm, such that our method can automatically spontaneously

TABLE 1. Details of the datasets.

Datasets	Clicks	Train_sesions	Test_sessions	Items	Average Length
YOOCHOOSE 1/4	8326407	5917746	55898	29618	5.71
YOOCHOOSE 1/64	557248	369859	55898	16766	6.16
DIGINETICA	982961	719470	60858	43097	5.12

organise hierarchical meanings and corresponding processing for an operation to have different consequences and influence ranges within a wide range, greatly enhance our model is building flexibility and compatibility.

7) OUTPUT RECOMMENDED RESULTS

In the first two parts, we can get the embedded vector for each session. We multiply the embedding vector v_i of each item $v_i \in \mathcal{V}$ within the consideration range by its attention vector $h_{L,S}$, calculate the score S of each candidate v_i, and then obtain the predicted output of the model through the softmax layer:

$$S = h_{L,S}^{T} v_i \tag{18}$$

$$\hat{y} = softmax(S) \tag{19}$$

In summary, we propose a GCNs structure with a 5-layer 1^{st}-order approximation as a short-time user preference transfer module and add ConvLSTM structure to enhance the simulation ability for the entire long-term process of preference transfer. The module of structural learning ability learns the overall behaviour pattern of the user at multiple levels, thus forming a complete user call behaviour session system that takes into account the long-term and short-term effects.

IV. EXPERIMENTAL DESIGN

In this part, we first describe the dataset and the parameter selection of the model used in the experiment. We also compared the different ways of connecting the network. Then, we compare GACOforRec with the baseline algorithm of several recommendation algorithms and a Graph Convolutional Networks model (ST-GCN) [30] that is closer to our model.

A. ENVIRONMENT SETTINGS

All experiments were completed in the following environment: Python3.6, Pytorch1.0, TITANXP GPUs. In the setting of the model parameters, we chose learning rate = 0.001 and batch size = 4 to train the model.

B. DATASETS

The essence of deep learning is the mining of data. The algorithm learns from a large amount of data, and explores the relationship between the features hidden behind it, and acquires the ability to recognise, regress, and generate. The emergence of ever-larger datasets can drive the development of deep learning. The dataset we conducted in this paper draws on the model SR-GNN proposed by Shu Wu in 2018.

The experiment is based on two real datasets, YOOCHOOSE and DIGINETICA. The specific data of the datasets are shown as TABLE 1.

C. BASELINE

Item-KNN (2001): Calculate the similarity between item A and item B by calculation, that is, find all users directly related to A and B, and calculate the evaluation bias. After the calculation, we get the k most similar items.

BPR-MF (2009): This method is based on matrix decomposition and optimises the pairwise ordered objective function by stochastic gradient descent.

GRU-Rec (2016): It is the use of cyclic neural networks to build user sequence models for session-based recommendations.

NARM (2017): Based on the RNN, the attention mechanism has been added. Based on the analysis of the sequential action of the RNN, the primary behaviour of the user is more closely concerned.

SR-GNN (2019): This model was proposed by ShuWu et al. in January 2019 to aggregate separated session sequences into graph structure data. Through the Graph Neural Network, the global session preference and local preference are comprehensively considered.

D. EVALUATION METRICS

In our experiments, we used the following two metrics to evaluate our model and compare it to the baseLine in IV-C.

Recall@20: It is an essential one in the recommended system evaluation index, indicating the proportion of correctly recommended items in the first 20 items.

MRR@20: That is Mean Reciprocal Rank. It represents the average of the peer-level levels of the correctly recommended items. The MRR metric takes into account the order in which the rankings are recommended, with a more substantial MRR value indicating that the correct recommendation is at the top of the ranking list.

These two indicators describe several vital features that should be included in a recommendation system from different angles. In addition to the recommendation, the recommendation should be completed as soon as possible. This is also the focus of our model considering the timing relationship. Both of the above metrics are, and the larger the value, the better the model is.

V. EXPERIMENTAL RESULTS

A. COMPARISON WITH BASELINE

We compare the five main methods in GACOforRec and Baselines on Recall and MRR. The experimental results are shown in TABLE 2.

From the experimental results shown by TABLE 2, on the two datasets DIGINETICA and YOOCHOOSE1/64

TABLE 2. Comparison results with baseline algorithms.

Datasets	DIGINETICA		YOOCHOOSE 1/64		YOOCHOOSE 1/4	
MethodsMetric	Recall@20	MRR@20	Recall@20	MRR@20	Recall@20	MRR@20
BPR-MF	15.19	8.63	31.31	12.08	3.40	3.40
GRU-Rec	43.82	15.46	60.64	22.89	59.23	1.57
Item-KNN	28.35	9.45	51.60	12.08	52.31	21.70
SR-GNN	-	17.59	-	30.94	-	31.89
NARM	62.58	27.35	68.32	28.76	69.73	29.23
GACOforRec	**68.51**	**30.07**	**69.79**	**29.38**	**67.66**	**28.13**

TABLE 3. Performance of different network structures in the model.

NETWORK STRUCTURE	Recall@20
GCNs+LSTM	67.72
GCNs+GRU	66.53
GCNs+ConvLSTM	66.13
GCNs+ON-LSTM	67.92
GCNs+ON-LSTM+GCN	66.93
GCNs+ConvLSTM+GCNs+ON-LSTM	68.51

(especially DIGINETICA), our model performance is significantly better than the existing baseline algorithm. The indicator of the YOOCHOOSE1/4 dataset is slightly smaller because YOOCHOSE1/4's number of sessions for training and testing is much larger than 1/64. This also illustrates from another perspective that our model is more suitable for handling larger session datasets and does not perform well on small datasets.

B. ABLATION EXPERIMENTS

1) COMPARISON WITH NETWORK CONNECTION OF MODEL

We have tried different combinations of Graph Convolutional Networks and other recurrent neural networks TABLE 3 to find the best network structure. The results in TABLE 3 validate our previous analysis. We believe that a single GCN is not sufficient to describe the entire application scenario, nor can it handle unstructured data relationships between users and items. The experimental results obtained according to our model framework (GACOforRec) are better enough to describe our recommended environment fully.

2) EXPERIMENTAL ANALYSIS OF LONG-SHORT ATTENTION MECHANISM

We propose a pair of attention mechanisms (Long-Short Attention Mechanism) in the model. Short-Attention can maintain strong contextual relevance in a short time interval. In contrast, Long-Attention allows our model to be more flexible when dealing with items that involve a more extensive time range.

We demonstrated the need for the Long-Short Attention Mechanism on the DIGINETICA and YOOCHOOSE 1/64 datasets. The result is shown in Fig. 3.

We also give the reason for the choice of the parameter k in the Long-Attention Mechanism. In Fig. 4, We can see it intuitively that the parameter k directly affects the time range in our attention mechanism when $k < 5$, the performance of

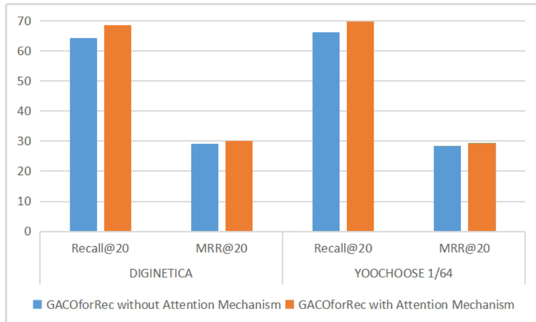

FIGURE 3. Performance comparison of attention mechanism.It proves the existence of attention mechanisms.

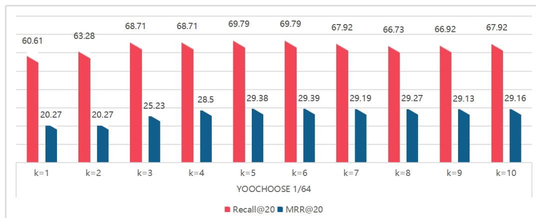

FIGURE 4. Parameter k selection in Long-Attention. In the model, different K values bring different performance.We select the relatively optimal K value through these results.

the model is significantly improved. When $k > 5$, there is no significant performance improvement.

3) COMPARISON WITH SPATIAL TEMPORAL GRAPH CONVOLUTIONAL NETWORKS (ST-GCN)

The technique of dynamic bone-based motion recognition method ST-GCN [30] proposed in 2018 has attracted our attention. The algorithm models the dynamic bone based on the time series representation of the human joint position and expands the graph convolution into a spatiotemporal Graph Convolution Networks to capture this temporal and spatial variation. This comparison test is since the focus on temporal and spatial information in this study is similar to ours. We compared GACOforRec with ST-GCN. To highlight the role of "time and space" in the session recommendation, we additionally added GACOforRec with the ConvLSTM layer added to the comparison test. The experimental results are shown in Fig. 5 and Fig. 6.

C. EXPERIMENTAL DISCUSSION

Based on the experimental results, we can get the following three conclusions:

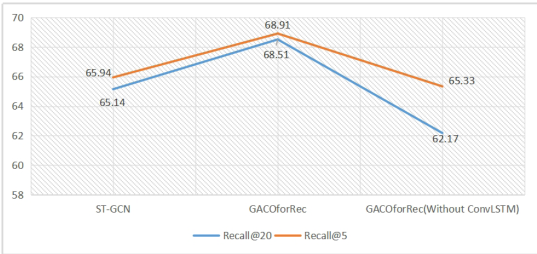

FIGURE 5. Performance comparison of Recall@20 and Recall@5.

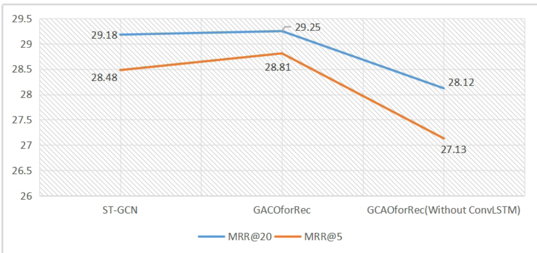

FIGURE 6. Performance comparison of MRR@20 and MRR@5.

1) GCN IS MORE SUITABLE FOR THE UNSTRUCTURED APPLICATION ENVIRONMENT OF THE RECOMMENDATION SYSTEM

The processing capacity of GCNs for non-euclidean domain data is far better than that of Recurrent Neural Network and Convolutional Neural Network. Through propagation, the nodes in the graph are connected in the spatial domain, making the connection between objects closer and closer.

2) GCN WORK BETTER WITH OTHER NETWORKS THAT IMPLEMENT SPECIFIC FUNCTIONS

In session-based recommendations, time and space factors are critical and must be considered. It can be seen that the GACOforRec with the ConvLSTM layer removed has a decrease in both Recall and MRR.

3) SESSION-BASED RECOMMENDATIONS ARE REQUIRED FOR THE CHOICE OF GRAPH CONVOLUTIONAL NETWORKS

Not every Graph Convolutional Networks model can perform well in session recommendations. It can be seen that although ST-GCN performs well on video data, its performance has declined in a session-based environment. Experiments have shown that our model works better in session recommendations.

VI. CONCLUSION

Our proposed model, GACOforRec, is a recommendation model that comprehensively focuses on users' short-term preference and volatility discrete preference. Through a lot of experiments, our model not only considers complex data forms and network structures but also captures both temporal and spatial information. Not only that, but our model also enables unsupervised, hierarchical learning of potential user preferences. Experimental results show that our model has good performance.

We always believe that graph structures (especially Graph Convolutional Neural Networks) are the best way to model users and objects in a recommendation system. In the future, we will focus on the processing of non-independent and identically distributed data of multimodal data by other Graph Convolutional Networks models, or apply more advanced techniques for feature extraction of spatiotemporal information to our existing models. Improve the ability of the model to mine and analyse small and biased datasets [2].

REFERENCES

[1] D. Beck, G. Haffari, and T. Cohn, "Graph-to-sequence learning using gated graph neural networks," 2018, *arXiv:1806.09835*. [Online]. Available: https://arxiv.org/abs/1806.09835

[2] S. Bonner and F. Vasile, "Causal embeddings for recommendation," 2017, *arXiv:1706.07639*. [Online]. Available: https://arxiv.org/abs/1706.07639

[3] L. Cai, J. Xu, J. Liu, and T. Pei, " Integrating spatial and temporal contexts into a factorization model for POI recommendation," *Int. J. Geograph. Inf. Sci.*, vol. 32, no. 3, pp. 524–546, 2018.

[4] C. Wu, M. Yan, and L. Si, "Session-aware information embedding for E-commerce product recommendation," 2017, *arXiv:1707.05955*. [Online]. Available: https://arxiv.org/abs/1707.05955

[5] P. Christodoulou, S. P. Chatzis, and A. S. Andreou, "A variational latent variable model with recurrent temporal dependencies for session-based recommendation (VLaReT)," in *Advances in Information Systems Development*. 2018.

[6] W. Dong and M. Chi, "Long short-term memory with quadratic connections in recursive neural networks for representing compositional semantics," *IEEE Access*, vol. 5, pp. 16077–16083, 2017.

[7] F. Eskandanian and B. Mobasher, "Detecting changes in user preferences using Hidden Markov models for sequential recommendation tasks," 2018, *arXiv:1810.00272*. [Online]. Available: https://arxiv.org/abs/1810.00272

[8] D. Gao, Y. Tong, J. She, T. Song, L. Chen, and K. Xu, "Top-*k* team recommendation and its variants in spatial crowdsourcing," *Data Sci. Eng.*, vol. 2, no. 2, pp. 136–150, Jun. 2017.

[9] R. He and J. Mcauley, "Fusing similarity models with Markov chains for sparse sequential recommendation," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2017, pp. 191–200.

[10] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel recurrent neural network architectures for feature-rich session-based recommendations," in *Proc. ACM Conf. Recommender Syst.*, 2016, pp. 241–248.

[11] D. Jannach and M. Ludewig, "When recurrent neural networks meet the neighborhood for session-based recommendation," in *Proc. 11th ACM Conf. Recommender Syst.*, 2017, pp. 306–310.

[12] L. Jing, P. Ren, Z. Chen, Z. Ren, and J. Ma, "Neural attentive session-based recommendation," 2017, *arXiv:1711.04725*. [Online]. Available: https://arxiv.org/abs/1711.04725

[13] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: https://arxiv.org/abs/1609.02907

[14] R. Li, M. Tapaswi, R. Liao, J. Jia, and S. Fidler, "Situation recognition with graph neural networks," 2017, *arXiv:1708.04320*. [Online]. Available: https://arxiv.org/abs/1708.04320

[15] T. Li, A. Liu, and C. Huang, "A similarity scenario-based recommendation model with small disturbances for unknown items in social networks," *IEEE Access*, vol. 4, pp. 9251–9272, 2016.

[16] H. Liu, X. Kong, X. Bai, W. Wang, T. M. Bekele, and F. Xia, "Context-based collaborative filtering for citation recommendation," *IEEE Access*, vol. 3, pp. 1695–1703, Oct. 2015.

[17] 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.*, 2018, pp. 1831–1839.

[18] M. Ludewig and D. Jannach, "Evaluation of session-based recommendation algorithms," *User Model. User-Adapt. Interact.*, vol. 28, pp. 331–390, Dec. 2018.

[19] T. Neammanee and S. Maneeroj, "Time-aware recommendation based on user preference driven," in *Proc. IEEE Comput. Softw. Appl. Conf.*, Jul. 2018, pp. 26–31.

[20] L. Qiang, W. Shu, W. Liang, and T. Tan, ''Predicting the next location: A recurrent model with spatial and temporal contexts,'' in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016.

[21] X. Shi, Z. Chen, W. Hao, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, ''Convolutional LSTM network: A machine learning approach for precipitation nowcasting,'' in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[22] T. Stepan, J. M. Morawski, S. Dick, and J. Miller, ''Incorporating spatial, temporal, and social context in recommendations for location-based social networks,'' *IEEE Trans. Comput. Social Syst.*, vol. 3, no. 4, pp. 164–175, Dec. 2016.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, ''Attention is all you need,'' 2017, *arXiv:1706.03762*. [Online]. Available: https://arxiv.org/abs/1706.03762

[24] J. B. P. Vuurens, M. Larson, and A. P. de Vries, ''Exploring deep space: Learning personalized ranking in a semantic space,'' in *Proc. Workshop Deep Learn. Recommender Syst.*, 2016, pp. 23–28.

[25] H. Wang and W. Liang, ''Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks,'' 2017, *arXiv:1704.02581*. [Online]. Available: https://arxiv.org/abs/1704.02581

[26] W. Wang, H. Yin, C. Ling, Y. Sun, S. Sadiq, and X. Zhou, ''Geo-SAGE: A geographical sparse additive generative model for spatial item recommendation,'' 2015, *arXiv:1503.03650*. [Online]. Available: https://arxiv.org/abs/1503.03650

[27] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, ''Session-based recommendation with graph neural networks,'' in *Proc. 23rd AAAI Conf. Artif. Intell. (AAAI)*, 2019, pp. 346–353. [Online]. Available: http://arxiv.org/abs/1811.00855

[28] S. Wu, Y. Tang, Y. Zhu, X. Xie, and T. Tan, ''Session-based recommendation with graph neural networks,'' 2018.

[29] W. Wu, R. Zhang, and L. Liu, ''A personalized network-based recommendation approach via distinguishing user's preference,'' *Int. J. Mod. Phys. B*, vol. 33, no. 6, 2019, Art. no. 1950029.

[30] S. Yan, Y. Xiong, and D. Lin, ''Spatial temporal graph convolutional networks for skeleton-based action recognition,'' 2018, *arXiv:1801.07455*. [Online]. Available: https://arxiv.org/abs/1801.07455

[31] Y. K. Tan, X. Xu, and Y. Liu, ''Improved recurrent neural networks for session-based recommendations,'' 2016, *arXiv:1606.08117*. [Online]. Available: https://arxiv.org/abs/1606.08117

[32] Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, and T.-Y. Liu, ''Sequential click prediction for sponsored search with recurrent neural networks,'' in *Proc. 8th AAAI Conf. Artif. Intell.*, 2014, pp. 1369–1375.

[33] P. Zhao, H. Zhu, Y. Liu, Z. Li, J. Xu, and V. S. Sheng, ''Where to go next: A Spatio-temporal LSTM model for next POI recommendation,'' 2018, *arXiv:1806.06671*. [Online]. Available: https://arxiv.org/abs/1806.06671

[34] C. Zhou, J. Bai, J. Song, X. Liu, Z. Zhao, X. Chen, and J. Gao, ''ATRank: An attention-based user behavior modeling framework for recommendation,'' 2017, *arXiv:1711.06632*. [Online]. Available: https://arxiv.org/abs/1711.06632

[35] Y. Zhu, H. Li, Y. Liao, B. Wang, Z. Guan, H. Liu, and D. Cai, ''What to do next: Modeling user behaviors by time-LSTM,'' in *Proc. IJCAI*, 2017, pp. 3602–3608.

MINGGE ZHANG received the B.Sc. degree in administration from the Tianjin University of Science and Technology, in 2018. She is currently pursuing the M.S. degree in software engineering with the Qilu University of Technology (Shandong Academy of Sciences). Her research interests include deep learning, interpretable reasoning, and recommendation systems.

ZHENYU YANG received the B.S. degree in computer application technology from the Qilu University of Technology (Shandong Academy of Sciences), in 2007. He is currently pursuing the M.S. degree in control engineering and theory with the China University of Mining and Technology. He is currently an Associate Professor with the Qilu University of Technology (Shandong Academy of Sciences) and the Deputy Director of the Software Integration Institute. His research interests include artificial intelligence, knowledge management, and information integration.