Cross-Session Aware Temporal Convolutional Network for Session-based Recommendation

Rui Ye*, Qing Zhang*, Hengliang Luo
Meituan
Beijing, China
{yerui03, zhangqing31, luohengliang}@meituan.com

2020 International Conference on Data Mining Workshops (ICDMW) | 978-1-7281-9012-9/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICDMW51313.2020.00039

***Abstract*—Recent advancements in Graph Neural Networks (GNN) have achieved promising results for the session-based recommendation, which aims to predict a user's actions based on anonymous sessions. However, existing graph-structured recommendation methods only focus on the internals of a session and neglect cross-session effect which contains valuable complement information for more accurately learning the taste of the user in the current session. Meanwhile, the graph structure lacks the sequential position information so that different sequential sessions can be constructed as the same graph, inevitably limiting its capacity of obtaining an accurate vector of a session representation. In order to solve the above limitations, we propose Cross-session Aware Temporal Convolutional Network (CA-TCN) model. For the cross-session aware aspect, CA-TCN builds a global-item graph and a session-context graph to model cross-session influence on both items and sessions. *Global-item graph* explores the global cross-session influence on items by building relevant item connections among all sessions. *Session-context graph* explores the complex cross-session influence on sessions by building the connections between the current session and other sessions with similar user intents and behavioral patterns as the current session. And, we connect items and sessions with hierarchical item-level and session-level attention mechanism. Besides, compared with the GNN, TCN can perform convolution operation on multi-hops items and maintain sequence information in the process of convolution. Extensive experiments on two real-world datasets show that our method outperforms state-of-the-art methods consistently.**

***Index Terms*—Cross-session aware, Session-based recommendation, Temporal convolutional network, Hierarchical attention**

I. Introduction

In the era of big data, Recommender Systems play an increasingly important role as infrastructural facilities for users to select interesting information and alleviate the problem of information overload in various application domains. Conventional recommendation methods are often based on explicit user identification, while user identification maybe not available in some application domains. To address this problem, session-based recommendation is proposed to predict a user's next action based on the user's behavioral sequence in the current session without any user identification [1]. As a session is an items transition sequence, the session-based recommendation can be naturally regarded as a sequence learning problem. In the early phase, the traditional sequence methods, such as the Markov chain, are developed to predict the user's next action based on the last actions. But the combination of the past interactions is ignored, which may limit the accuracy of recommendation. In recent years, we have seen significant developments in session-based recommendation, especially the development of deep learning methods that are capable of learning sequential data of sessions. [2] [3] [4] achieve promising results, by applying Recurrent Neural Networks (RNN) [5] for session-based recommendation instead of traditional Markov chains and matrix factorization methods [6] [7] [8]. The work GRU4RC is the first attempt to propose a recurrent neural network approach, then on the basis of GRU4RC, GRU4RC+ takes data augmentation and a temporal shift of user behavior into consideration. HATE [9] proposes a hierarchical attentive transaction embedding model to integrate both item embedding and transaction embedding and [10] proposes a framework to integrate both explicit and hidden item dependencies. NARM proposes a concept of global and local session representation based on RNN to capture users' sequential behavior and main purposes simultaneously. STAMP also captures users' general interests and current interests, by employing simple MLP networks and an attentive net. With the success of Transformer [11] in text understanding, Bert4Rec [12] seeks to use a Bert model to explore the sequence pattern for session-based recommendation.

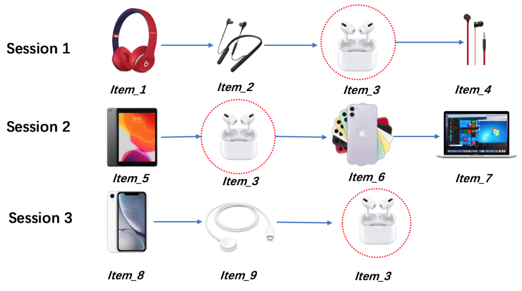

Fig. 1. The toy example for cross-session items

More recently, SR-GNN [13] and GC-SAN [14] adopt the GNN [15] to explore rich transitions among items by building each session as a graph structure. The work SR-GNN proposes to use the graph neural network in session-based recommendation at first. Then GC-SAN combines the GNN and self-attention adopted in Transformer to capture complex transitions of items. Although the graph-structured neural

* Corresponding author.

2375-9259/20/$31.00 ©2020 IEEE
DOI 10.1109/ICDMW51313.2020.00039

network models have become the state-of-the-art solutions, they still have some limitations.

First, almost existing session-based recommendation methods only focus on the internals of a session and neglect cross-session information which is vital for improving recommendation performance. CSRM [16] is the only work incorporating collaborative information from the latest sessions to enrich the representation of the current session. Moreover, CSRM computes the similarities between the current session and the latest m sessions to extract collaborative information. But it only focuses on the other session influence on the session-level and unfortunately ignores the cross-session influence on the item-level. In the task of session-based recommendation, the item should be the minimum granularity, as the item is the unit of session, meanwhile item is very likely to appear in multiple sessions which items in other sessions also have a relevant effect on the current item.

In Fig. 1, we take item_3 Airpods in Session 3 as an example, item_3 appears in three sessions. Existing methods only focus on the effect of item_9 in the current session 3 on item_3 and neglect the item effect in other sessions. In fact, item_3 should be affected by two aspects, one comes from the item_9 in the current session 3, and the other comes from item_2, item_4, item_5, and item_6 in other sessions. For Session 1, the user has the intention of buying headphones to compare them in the same category. So the item_2, item_4 should contribute a cross-session effect on item_3 with the same category. For session 2, the user focuses on the Apple brand product so item_5 and item_6 should contribute cross-session affect on item_3 with the same brand. From the above observations, the cross-session item influence is essential for better inferring item global representation. Meanwhile, cross-session influence in session-level also plays a vital role for more accurately predicting the user's next action in the current session. Because different sessions may have similar user intents and behavioral patterns.

Second, the graph structure lacks the sequence order information because the graph structure views the same item which appears at different time steps in the current session as the same node so that different sequential sessions can be constructed as the same graph. As in Fig. 2, the sequential session S1: $v_i \rightarrow v_j \rightarrow v_i \rightarrow v_k \rightarrow v_j \rightarrow v_k$, and S2: $v_i \rightarrow v_j \rightarrow v_k \rightarrow v_j \rightarrow v_i \rightarrow v_k$, have the identical graph structure even though two sequences are different from each other, thus inevitably limiting model capacity to obtain accurate session vectors. Furthermore, for the session graph construction in SR-GNN, the edge only connecting two adjacent items, means that only the last clicked item is the directly connected first-order neighbor of the current item as in Fig. 3. The long-term dependency information needs to stack multi-layer to acquire, but the over-smoothing character of graph neural networks can't support the operation. Even though items that appear in the same session are not be clicked in successive, but they also have a certain connection. GNN has limited power for the long-term dependency of sequence. Conversely, for the TCN [17] model, the causal convolution makes items within receptive fields of the current item can be directly convoluted as the first-order neighbors and with the characteristics of dilated convolution, more distant items can be directly affected as the first-order neighbor.

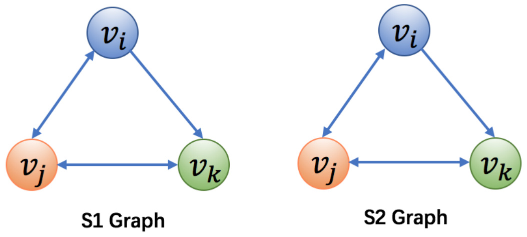

Fig. 2. Session Graph

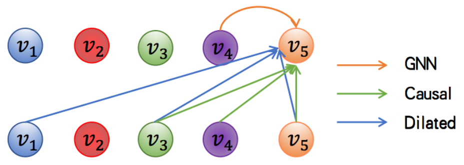

Fig. 3. The limits of graph structure for sequence learning

We propose CA-TCN to overcome the above limitations. The cross-session information has two types: 1. Cross-session items that exist in other session and have the similar property as the current item; 2. Session-context that reflects similar user intents and behavioral patterns as the current session. For the item level, we build a global cross-session item directed graph and adopt GNN to explore the rich cross-session influence on each item. For the session level, we construct a cross-session graph where the existence of an edge is determined by a similarity score between two session representations. The TCN has a natural advantage for sequence learning with causal and dilated convolution to overcome the limitations (e.g., parallelization of RNN-based models and lacking the sequence order and long-term dependency information of graph-based models). TCN is a sequence convolution model, and the sequence information will not be lost during the convolution process, that is, the same item at different time steps has a different representation. Moreover, TCN also can be regarded as the multi-hops graph neural network since TCN can make use of the direct influence among the items even though they not are clicked in succession. Besides, item-level and session-level hierarchical attention are applied to get the optimal combination of multiple items and sessions in a hierarchical manner, which enables the learned session representation to better capture the rich information of complex session structure.

The main contributions of this work are summarized as follows.

- To our best knowledge, CA-TCN is the first model to explore the cross-session influence on item-level and session-level simultaneously. Furthermore, CA-TCN constructs a cross-session item graph and session-context graph to take advantage of related items from other sessions and interaction among different sessions to enhance the recommendation performance.

- We propose to apply TCN to solve the limits of the graph-structure models which lack the sequence order and long-term dependency. TCN can perform the convolution operation on long-term items and maintain sequence information in the process of convolution.
- We propose item-level and session-level hierarchical attention. We can take the importance of items and sessions into consideration simultaneously, benefiting from such hierarchical attention.
- We conduct extensive experiments on benchmark datasets. Our experimental results show the effectiveness and superiority of CA-TCN, comparing with the state-of-the-art methods.

II. Related Work

The traditional recommender systems (e.g., the content-based and collaborative filtering methods), model the user-item interactions, and can only capture a user's general preference. In contrast, session-based recommendation treats the user-item interactions as a sequence and takes the sequential dependencies into account to predict the next action of a user for more accurate recommendation [18] [19].

Markov Chain methods Due to the highly practical value of session-based recommendation, there are some traditional methods such as Markov Chain-based methods to be developed, even though they are not originally designed for session-based recommendation. The Markov Chain-based approaches regard each session as a Markov chain and a transition matrix is estimated that gives the probability of the next action based on the last action of the user. FMPC presents a method bringing both matrix factorization and Markov chain together to learn the taste of a user. However FMPC only considers the pair transaction, to capture more distant items, the high-order Markov chain approaches are also developed.

Collaborative filtering methods. Collaborative filtering is a general and widely used approach for recommendation systems. The collaborative filtering methods can fall into two categories: K-Nearest Neighbor (KNN-based) methods [20] and model-based methods [16]. The KNN-based methods achieve recommendations by finding top-k users or items. KNN-based methods can be extended for session-based recommendation by recommending items that are most similar to the last item of the current session. Recently, KNN-RNN [20] explores to combine an RNN model with a co-occurrence-based KNN model to extract sequence information in session-based recommendations. For the model-based method, CSRM [16] takes collaborative information contained in the latest m neighborhood sessions into account for session-based recommendations to enhance the performance.

Deep-learning based methods. Motivated by the powerful representation learning capability and remarkable success of deep-learning, RNN is an intuitive choice to capture the complex intra-session dependency with its strength in handling sequential data. GRU4Rec [4] utilizes the Gated Recurrent Unit (GRU) as a special form of RNN to learn the long-term item dependency to predict the next action in a session. A series of improvements, extensions have been proposed based on RNN. Some works further extend RNN with attention and memory mechanism [2]. NARM explores a hybrid encoder with an attention mechanism to model the user's sequential behavior and main purpose in the current session. Bert4Rec [12] and [21] both utilize the Transformer to model the user behavior sequence. Because human behaviors are complex, IntNet [22] is designed to model the user actions driven by different intentions and MCPRN [23] develops modeling multi-purpose sessions for next item recommendations via mixture-channel purpose routing networks. More recently, there has been a surge of methods that rely on the graph structure. SR-GNN [13] first proposes to map each session as a graph structure and utilizes GNN to model complex transitions of items. GC-SAN [14] further extends GNN with self-attention mechanism [11], leading to state-of-the-art results. And HierTCN [24] adopts TCN to make recommendations based on users' sequential multi-session interactions.

Novelty. Our model has significant differences with existing methods. On the one hand, we explore the cross-session information on item-level and session-level to promote recommendation performance. The differences from other collaborative filtering methods are two-fold: 1. CA-TCN takes both cross-session items and session-context information into consideration simultaneously, while CSRM [16] only considers session-context information; 2. Our model builds a cross-session global item graph and session-context graph to explore the complex cross-session effects via GNN. On the other hand, compared with the RNN-based and GNN-based model, the CA-TCN has the satisfying power to overcome the limitations (e.g., parallelization of RNN-based models and lacking the sequence order and long-term dependency information of graph-based models). Compared with HierTCN, HierTCN needs to know the corresponding user ID of the session, however, the session-based task doesn't use any supplementary information.

III. Method

We firstly formulate the task of session-based recommendation and then explain the cross-session concept on item level and session level, lastly describe the architecture of our model in detail as shown in Fig. 4.

Let $V = \{v_1, v_2, ..., v_{|V|}\}$ denotes a set of all unique items involved in all sessions. Each session, the $s = \{v_{s,1}, v_{s,2}, ..., v_{s,n}\}$, consists of a sequence of actions where $v_{s,i} \in V$ represents a clicked item v_i of a user within the session s. The goal of the session-based recommendation is based on the previous click sequence $\{v_{s,1}, v_{s,2}, ..., v_{s,n}\}$ to predict the next click $v_{s,n+1}$ by recommending top-K items from items set V.

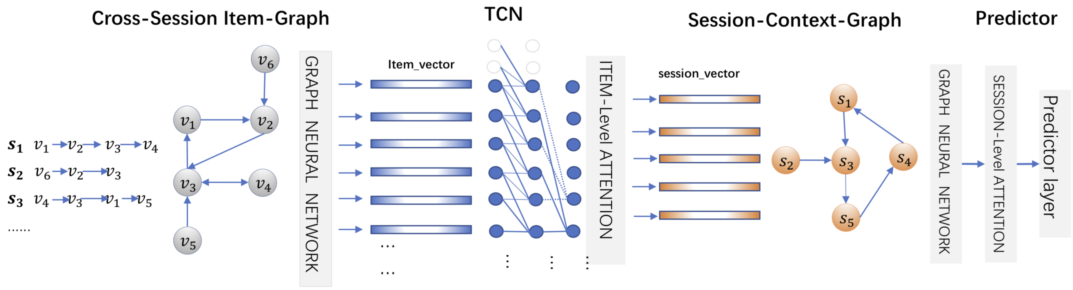

Fig. 4. The overview of proposed CA-TCN architecture.

A. *Cross-Session Item Graph*

In the first phase, we construct the cross-session item directed graph G_{item} where nodes are all unique item $v_i \in V$and $(v_{s,i}, v_{s,i+1})$ as an edge which represents a user clicks item $v_{s,i+1}$ after $v_{s,i}$ in the session s. In comparison with existing methods, the graph G_{item} is capable of building the connections between items that appear among all sessions, so G_{item} incorporates the cross-session and global item effect in addition to internal items of the current session. The key of cross-session influence on item level is that G_{item} can construct cross-session items link even though they do not appear in the same session or not be clicked by the same user as in Fig. 4.

In order to make full use of the all session information on item level, we take item click order and the number of item occurrences into consideration. For the item click order, we build the edge with the direction in cross-session item graph G_{item}, correspondingly we build adjacency matrix A_{in} and A_{out} to model the incoming and outgoing connections. On the basis of the adjacency matrix, we assign the different weights to the edge according to the number of item occurrences to produce the weight matrix called $Weight_{in}$ and $Weight_{out}$. For example, the item i has connection with j and g, if i and j jointly appear 100 times but i and g co-occur 10 times. In this sense, the connection between (i, j) and (i, g) should be different. By assigning different weight, the items which have more number of item occurrences play a greater role, and vice versa, avoiding the noise effect.

Next, we develop GNN to explore the complex cross-session influence on item_level. GNN embeds items into an unified embedding space $\mathbf{V} \in R^{m*dim}$ where dim is the vector dimension.

$$\mathbf{V}^{l+1} = \sigma(D^{-1}(Weight_{in}\mathbf{V}^l + Weight_{out}\mathbf{V}^l + \mathbf{V}^l)W^l) \quad (1)$$

where σ is the activation function (e.g., Sigmoid, Relu), D is the degree matrix of the graph G_{item} which is the sum of incoming degree and outgoing degree, W denotes the parameter matrix and l is the l-th layer of GNN.

B. *Temporal Convolutional Neural Network*

We utilize TCN to model session sequence and extract its local and global information with the input of global item vectors. For each session sequence $s = \{v_{s,1}, v_{s,2}, ..., v_{s,n}\}$ and the corresponding input item vectors $[\mathbf{v}_{s,1}, \mathbf{v}_{s,2}, ..., \mathbf{v}_{s,n}]$, we conduct the TCN convolution operator F on each item vector $\mathbf{v}_{s,i}$:

$$F(\mathbf{v}_{s,i}) = (\mathbf{v}_{s,i} *_d f) = \sum_{j=0}^{k-1} f(j) \cdot \mathbf{v}_{s,i-d \cdot j} \quad (2)$$

where f is the filter, where d is the dilation factor, k is the filter size. $n - d * j$ accounts for the direction of the past, because causal convolution makes no information leakage from the future into the past. In the process of conducting convolution on item $v_{s,i}$ as in Fig. 4, the items within receptive fields can be directly convoluted which can be seen the first-order neighborhood in the graph structure and the dilated convolution allows for receptive fields exponential to the number of layers to obtain the user's long-term interest.

Then, we use the final output $F(\mathbf{v}_{s,n})$ of TCN as the local representation $\mathbf{s}^{local}$ of the current session $\mathbf{s}$ to correctly capture the user' current interests:

$$\mathbf{s}^{local} = F(\mathbf{v}_{s,n}) \quad (3)$$

Besides, we consider the global representation $\mathbf{s}^{global}$ of the session $\mathbf{s}$ by aggregating all item vectors with the item-level attention to capture user's general interest. The item-level attention assigns different weights among items in the current session:

$$\alpha_{s,i} = softmax(\sigma(\mathbf{W}_1 F(\mathbf{v}_{s,n}) + \mathbf{W}_2 F(\mathbf{v}_{s,i}))) \quad (4)$$

where $\alpha_{s,i}$ is the weight of item i in the current session, $\mathbf{W}_1$, $\mathbf{W}_2$ transform $\mathbf{v}_{s,n}$ and $\mathbf{v}_{s,i}$ into latent spaces, respectively. Then we can capture the current session intent by adaptively focusing on more important items:

$$\mathbf{s}^{global} = \sum_{i=1}^{n} \alpha_{s,i} \cdot F(\mathbf{v}_{s,i}) \quad (5)$$

C. *Session-Context Graph*

The local and global representations of sessions only focus on the current session, while lacking the cross-session information. To address this problem, we build a session-context graph $G_{session}$ to take complex relations among sessions into account. In the graph $G_{session}$, a node represents a session and a non-trivial problem that we need to discuss is how to identify graph edges. Especially, given the current session s_i, we firstly take the latest m sessions as the candidate neighborhood sessions. To further obtain the most similar K neighborhoods, we compute the representation similarity between current session and candidate sessions and use approximate KNN graph algorithm [25] to determine neighbors of a session node, which is connected to its top-k similar session nodes with edges. The similarity function as follows:

$$sim_{ij} = sim(s_i, s_j) = \frac{\mathrm{s}_i^{global}\mathrm{s}_j^{global}}{||\mathrm{s}_i^{global}||||\mathrm{s}_j^{global}||} \tag{6}$$

After constructing the session-context graph $G_{session}$, we develop session-level attention GAT [26] to incorporate the neighborhood nodes effects. Different from existing GAT, the session-level attention GNN will take the session similarity into the computation. In each layer of session-level attention graph neural network, the edge weight coefficient can be computed as :

$$\alpha_{i,j} = \frac{exp(LeakyReLU(\mathbf{a}^T[\mathbf{W}\mathrm{s}_i^{global}||\mathbf{W}\mathrm{s}_j^{global}||sim_{ij}]))}{\sum\limits_{k \in N_i} exp(LeakyReLU(\mathbf{a}^T[\mathbf{W}\mathrm{s}_i^{global}||\mathbf{W}\mathbf{s}_k^{global}||sim_{ik}]))} \tag{7}$$

where N_i denotes the neighborhoods of session s_i in $G_{session}$, $||$ denotes the concatenation operator, **a** is the transformation vector, W is the paramater matrix. After that, the session-context representation of s_i can be computed by aggregating the neighborhood representations:

$$\mathbf{s}_i^{cross} = \sum_{j \in N_i} \alpha_{i,j}\mathbf{s}_j^{global} \tag{8}$$

After applying the graph neural network, we acquire the cross-context session representations which contain the cross-session influence.

D. Recommendation Decoder

To better predict the user's next action, we utilize a fusion function to combine the current interest, long-term preference, and cross-session information to get the final representation of a session.

$$f = \sigma(\mathbf{W}_l\mathbf{s}^{local} + \mathbf{W}_g\mathbf{s}^{global} + \mathbf{W}_s\mathbf{s}^{cross}) \tag{9}$$

$$\mathbf{s}_{final} = f \cdot [\mathbf{s}^{local}||\mathbf{s}^{global}] + (1 - f) \cdot \mathbf{s}^{cross} \tag{10}$$

where $\mathbf{W}_l$, $\mathbf{W}_g$ and $\mathbf{W}_s$ are the parameter matrix. Finally, we compute the probability for each candidate item v_i to predict the user next action.

$$\hat{y}_i = softmax(\mathbf{s}_{final}^T\mathbf{v}_i) \tag{11}$$

The objective function is defined as the cross-entropy of the prediction $\hat{y}$ and the ground truth y. y denotes the one-hot encoding vector of the ground truth item and m is the number of sessions. It can be written as follows:

$$L(\hat{y}) = -\sum_{i=1}^{m} y_i log(\hat{y}_i) + (1 - y_i)log(1 - \hat{y}_i) \tag{12}$$

IV. Experiment

In this section, we firstly describe the settings of the experiment. And then, we compare CA-TCN with the start-of-the-art baselines and analyze the results. Finally, we explore the effect of different components in CA-TCN.

A. Datasets and Evaluation

To evaluate the performance of the proposed CA-TCN, we employ two well-known benchmark datasets, namely, *Yoochoose*[1] and *Diginetica*[2]. The Yoochoose dataset is obtained from the RecSys Challenge 2015, which contains a stream of user clicks on an e-commerce website within 6 months. The Diginetica dataset comes from CIKM Cup 2016, where only its transactional data is used. For a fair comparison, following [13], we filter out all sessions of length 1 and items appearing less than 5 times in both datasets. Statistical details of datasets are shown in Table. I.

We adopt the widely used evaluation metric P@20 (Precision) and MRR@20 (Mean Reciprocal Rank) following [13].

P@20 (Precision) is widely used as a measure of predictive accuracy. P@20 represents the proportion of correctly recommended items amongst the top-20 items.

MRR@20 (Mean Reciprocal Rank) is the average of reciprocal ranks of the correctly-recommended items. The reciprocal rank is set to 0 when the rank exceeds 20. The MRR measure considers the order of recommendation ranking, where large MRR value indicates those correct recommendations at the top of the ranking list.

B. Baselines and Settings

All parameters are initialized by using a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. The mini-batch Adam optimizer is exerted to optimize parameters, where the learning rate is set to 0.0005. The dimensionality of session vectors is 100 and batch_size is 100. In the session-context graph, the number of cross-session neighbors is 100. The hyper-parameters for baselines are the same as reported in their papers. We tune the hyperparameters on a validation set which is a random 10% subset of the training set.

The following state-of-the-art approaches are included in the comparison for recommendation performance:

- **POP** is a naive baseline that recommends the most popular items of the training set.
- **FPMC** [6] combines Markov chain and matrix factorization for next-basket recommendation.
- **GRU4Rec** [4] is an RNN-based baseline that uses RNN to model user sequences.
- **KNN-RNN** [20] combines an RNN model with a co-occurrence-based KNN model for recommendation.
- **NARM** [2] incorporates the attention mechanism to RNN based model to improve performance.
- **STAMP** [3] is capable of capturing users' general interests and current interests.

[1] http://2015.recsyschallenge.com/challege. html
[2] http://cikm2016.cs.iupui.edu/cikm-cup

TABLE I
Statistical details of datasets

Dataset	clicks	train	test	all item	avg.length
Yoochoose1/64	55,7248	36,9859	55,898	16,766	6.16
Yoochoose1/4	8,326,407	591,7746	55,898	2,9618	5.71
Diginetica	98,2961	719,470	60,858	43,097	5.12

TABLE II
Performance comparison on session-based recommendation, Results marked (*) taken from [13], Results marked (**) are from experiments based on the authors' GitHub repositories. GC-SAN is re-implemented based on the SR-GNN' Github repositories

Dataset	Yoochoose 1/64		Yoochoose 1/4		Diginetica	
Metrics(%)	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
POP*	6.71	1.65	1.33	0.3	0.89	0.20
FPMC*	45.62	15.01	-	-	26.53	6.95
GRU4Rec*	60.04	22.89	59.53	22.60	29.45	8.33
RNN-KNN	68.66	28.64	68.02	28.38	44.61	15.35
NARM*	68.32	28.63	69.73	29.23	49.70	16.17
STAMP*	68.74	29.67	70.44	30.00	45.64	14.32
SR-GNN*	70.57	30.94	71.36	31.89	50.73	17.59
SR-GNN**	69.57	29.67	70.41	29.80	49.97	16.53
CSRM**	69.95	29.79	70.31	29.74	50.89	16.59
GC-SAN	70.47	30.18	70.83	29.95	50.91	17.18
CA-TCN	**71.98**	**31.01**	**72.25**	**32.23**	**53.66**	**18.19**

- **SR-GNN** [13] adopts the graph neural network to model session sequences.
- **GC-SAN** [14] utilizes both graph neural network and self-attention mechanism to capture rich dependencies.
- **CSRM** [16] applies collaborative neighborhood information to the session-based recommendation.
- **CA-TCN (ca.exl.)** is a variant of CA-TCN, which only contains temporal convolutional neural network and cross-session information on item-level and session-level excluded.
- **CA-TCN (sc.exl.)** is a variant of CA-TCN with the cross-session item graph but session-context graph excluded.

C. Results and Analysis

Table. II shows that CA-TCN outperforms all other methods on all datasets. Specifically,

1) The deep learning methods are significantly better than traditional methods such as POP, FPMC. It proves that deep learning methods have better support for capturing the sequence information than traditional methods.
2) Among deep learning methods, STAMP and NARM outperform GRU4Rec, confirming the effectiveness of capturing short-term memory and the main intent in a session to improve recommendation performance. Besides, CSRM which utilizes collaborative filtering information based on RNN, also performs better than others RNN-based methods, supporting the argument that other session influence is beneficial for acquiring session representation. The graph neural network methods such as SR-GNN and GC-SAN outperform other deep learning methods, showing the power of graph neural network on feature extracting. Especially, TCN adopted in CA-TCN can be regarded as the multi-hops graph neural network with sequence position information.
3) CA-TCN achieves consistent improvement on all datasets. The advantages of CA-TCN are mainly reflected in two aspects, respectively are sequence learning and cross-session effect. On the aspect of sequence learning, CA-TCN is better than RNN-based and GNN-based methods, such as NARM and SR-GNN. Because the TCN model in CA-TCN has better support for capturing the long-term dependency and position information of sequence by convoluting more distant items with causal and dilated convolution.
4) On the aspect of cross-session effect, CA-TCN performs better than both RNN-KNN and CSRM which take collaborative filtering information into account. The reasons are that CA-TCN not only proposes cross-session item and session-context effects simultaneously but also develops the graph-structure neural network to capture the complex nonlinear cross-session interactions.
5) We conducted experiments with different numbers of layers for GNN and TCN, both GNN and TCN with one layer have the best results. Because in our case, CA-TCN composes GNN and TCN to realize the cross-session and inner-session influence cooperatively, meanwhile each component has achieved item smoothness.

D. Influence of each Component

We conduct the ablation experiments to evaluate the influence of each component in CA-TCN, including TCN, cross-session item graph, and session-context graph. The Fig.5 depicts the experimental results.

- CA-TCN (ca.exl.) performs better than RNN-based and GNN-based methods, such as NARM and SR-GNN, even though no cross-session information is applied. Furthermore, CA-TCN (ca.exl.) outperforms GC-SAN which applies Transformer to model sequence pattern, showing that the TCN model is suitable for the session-based recommendation field.
- CA-TCN (sc.exl.) outperforms other baselines and CA-TCN (ca.exl.). The results indicate that the cross-session item graph is capable of incorporating complex cross-session item effects which are very beneficial for the

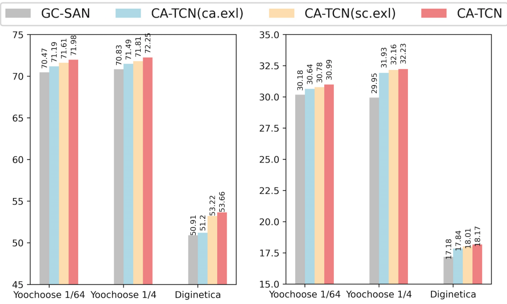

Fig. 5. Each component results on $P@20$ and $MRR@20$.

recommendation because the items in other sessions also have a relevant effect on the current item. CA-TCN is superior CA-TCN (sc.exl.). The only difference between them is the inclusion of a session-context graph. Therefore, it is evident that the cross-session information can improve the session representation.

- CA-TCN (ca.exl.) compared with SR-GNN** gains in performance obviously with the power of sequence learning. The G_{item} gains more improvement than $G_{session}$ since edges in G_{item} have strong correlation in the data.

These above observations confirm that CA-TCN leveraging TCN and cross-session information via GNN and hierarchical attention leads to substantial performance improvements in session-based recommendations.

V. Conclusions

In this paper, we propose a novel architecture "cross-session aware temporal convolution network (CA-TCN)" for the session-based recommendation. CA-TCN integrates cross-session influences on the item-level and session-level simultaneously to boost global item and session representations. Meanwhile, CA-TCN utilizes TCN to overcome the intrinsic limitation of losing sequence order and long-term dependency in graph-structured solutions. Besides, CA-TCN develops a hierarchical attention strategy to combine item and session representations with item-level and session-level attention. The experimental results validate that CA-TCN achieves state-of-the-art performance on benchmark datasets.

References

[1] S. Wang, L. Cao, and Y. Wang, "A survey on session-based recommender systems," *arXiv preprint arXiv:1902.04864*, 2019.

[2] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, "Neural attentive session-based recommendation," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1419–1428.

[3] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, "Stamp: short-term attention/memory priority model for session-based recommendation," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1831–1839.

[4] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," *arXiv preprint arXiv:1511.06939*, 2015.

[5] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model." in *INTERSPEECH*, T. Kobayashi, K. Hirose, and S. Nakamura, Eds. ISCA, 2010, pp. 1045–1048.

[6] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 811–820.

[7] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[8] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2016, pp. 549–558.

[9] S. Wang, L. Cao, L. Hu, S. Berkovsky, X. Huang, L. Xiao, and W. Lu, "Jointly modeling intra-and inter-transaction dependencies with hierarchical attentive transaction embeddings for next-item recommendation," *IEEE Intelligent Systems*, 2020.

[10] S. Wang and L. Cao, "Inferring implicit rules by learning explicit and hidden item dependency," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[12] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 1441–1450.

[13] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 346–353.

[14] C. Xu, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, F. Zhuang, J. Fang, and X. Zhou, "Graph contextualized self-attention network for session-based recommendation." in *IJCAI*, 2019, pp. 3940–3946.

[15] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel, "Gated graph sequence neural networks," *arXiv preprint arXiv:1511.05493*, 2015.

[16] M. Wang, P. Ren, L. Mei, Z. Chen, J. Ma, and M. de Rijke, "A collaborative session-based recommendation approach with parallel memory modules," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 345–354.

[17] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[18] X. Chen, H. Xu, Y. Zhang, J. Tang, Y. Cao, Z. Qin, and H. Zha, "Sequential recommendation with user memory networks," in *Proceedings of the eleventh ACM international conference on web search and data mining*, 2018, pp. 108–116.

[19] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. Orgun, "Sequential recommender systems: challenges, progress and prospects," *arXiv preprint arXiv:2001.04830*, 2019.

[20] D. Jannach and M. Ludewig, "When recurrent neural networks meet the neighborhood for session-based recommendation," in *RecSys '17*, 2017.

[21] Q. Chen, H. Zhao, W. Li, P. Huang, and W. Ou, "Behavior sequence transformer for e-commerce recommendation in alibaba," in *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*, 2019, pp. 1–4.

[22] S. Wang, L. Hu, Y. Wang, Q. Z. Sheng, M. Orgun, and L. Cao, "Intention nets: psychology-inspired user choice behavior modeling for next-basket prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 6259–6266.

[23] S. Wang, L. Hu, Y. Wang, Q. Z. Sheng, M. A. Orgun, and L. Cao, "Modeling multi-purpose sessions for next-item recommendations via mixture-channel purpose routing networks." in *IJCAI*, 2019, pp. 3771–3777.

[24] J. You, Y. Wang, A. Pal, P. Eksombatchai, C. Rosenburg, and J. Leskovec, "Hierarchical temporal convolutional networks for dynamic recommender systems," in *Proceedings of the 20th international conference on World wide web*, 2019, pp. 2236–2246.

[25] W. Dong, C. Moses, and K. Li, "Efficient k-nearest neighbor graph construction for generic similarity measures," in *Proceedings of the 20th international conference on World wide web*, 2011, pp. 577–586.

[26] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.