Temporal aware Multi-Interest Graph Neural Network for Session-based Recommendation

Qi Shen♠ *   2130777@tongji.edu.cn
Shixuan Zhu♠   2130768@tongji.edu.cn
Yitong Pang   1930796@tongji.edu.cn
Yiming Zhang   2030796@tongji.edu.cn
Zhihua Wei †   zhihua_wei@tongji.edu.cn
Tongji University

Editors: Emtiyaz Khan and Mehmet Gönen

Abstract

Session-based recommendation (SBR) is a challenging task, which aims at recommending next items based on anonymous interaction sequences. Despite the superior performance of existing methods for SBR, there are still several limitations: (i) Almost all existing works concentrate on single interest extraction and fail to disentangle multiple interests of user, which easily results in suboptimal representations for SBR. (ii) Furthermore, previous methods also ignore the multi-form temporal information, which is significant signal to obtain current intention for SBR. To address the limitations mentioned above, we propose a novel method, called *Temporal aware Multi-Interest Graph Neural Network* (TMI-GNN) to disentangle multi-interest and yield refined intention representations with the injection of two level temporal information. Specifically, by appending multiple interest nodes, we construct a multi-interest graph for current session, and adopt the GNNs to model the item-item relation to capture adjacent item transitions, item-interest relation to disentangle the multi-interests, and interest-item relation to refine the item representation. Meanwhile, we incorporate item-level time interval signals to guide the item information propagation, and interest-level time distribution information to assist the scattering of interest information. Experiments on three benchmark datasets demonstrate that TMI-GNN outperforms other state-of-the-art methods consistently.

Keywords: Session-based Recommendation, Graph Neural Network.

1. Introduction

Recommender system has become the basis to relieve the information overload problem. Most recommendation methods capture users' interest by modeling users' long-term preference for predicting their future interactions, e.g., collaborative filtering (He et al. (2017); Herlocker et al. (2004)) and neural network based models (Yu et al. (2016); Sedhain et al. (2015)). Differ from traditional recommendation tasks, in many practical scenarios, there is only a session available without access to user identification and historical interactions. This kind of task called session-based recommendation (SBR), aims to extract useful information as much as possible from limited data in current session.

---

† Both authors contributed equally to this research.
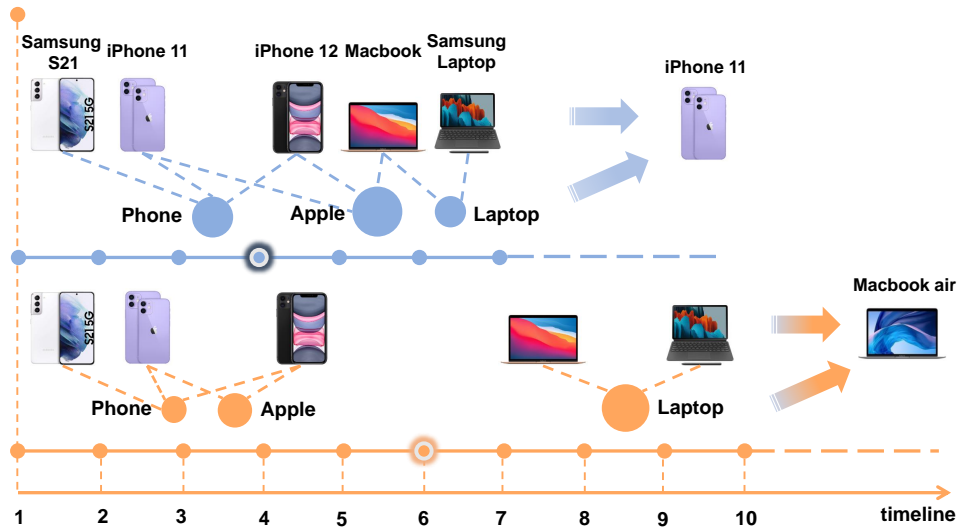† Corresponding author.

Figure 1: A toy example for two sessions with the same item sequence but different time intervals, where the circles connected to items denote the user's interests. Positions of the circles reflected the distribution of interest in the timeline, and the size indicates the importance of these interests.

Existing SBR methods mostly concentrate on modeling sequential information among items of current session by using Recurrent Neural Networks (RNNs) (Hidasi et al. (2015); Li et al. (2017)) or Graph Neural Networks (GNNs) (Wu et al. (2019); Pan et al. (2020)). However, these works simply regard a session as a short sequence with single intention, consider that the basic intention of user in a session usually remains the same, and try to capture user's current interest directly from the entire session. They overlook the fact that even in a relatively short-term session, the user's finer granular interests can be multiple in various views, drift over time, or interweave with item overlaps. Some studies (Zhou et al. (2018a); Li et al. (2019); Wang et al. (2020a); Lu et al. (2021); Cen et al. (2020); Chen et al. (2021b)) in other recommendation areas have verified the effectiveness of modeling user's multi-interest, but in SBR, the multi-interest methods have not been fully explored. As there is no interest disentangle mechanism in existing SBR methods, the mixture of major interests and minor interests may mislead the session representation learning so that it's hard to confirm users' true intentions and lacks of interpretability.

Moreover, the auxiliary temporal information is also ignored in session modeling. Intuitively, two typical temporal information of interaction sequences can be recognized for multi-interest modeling: (i) *item-level time interval signals* are the intervals between item transitions, which reflect the relatedness between adjacent items. Generally, the tight time interval indicates the higher relevance of items, while loose time interval may mean the drifting of interest. (ii) *interest-level distribution information* represents the interest distribution through virtual location and coverage scope in timeline, which is helpful to model the relation between item and interest from the time perspective. For the chunked interest

distribution in the session, the closer distance between items and interest factors in the timeline usually reveals the higher relatedness of them.

We illustrate this with an example in Figure 1, including two sessions with the same item sequence under different interaction time distributions. For the first session with short time intervals, as user's intention tends to be continuous over a short period of time, we can conclude that even item relatively far from current position may still represent user's strong interest. So user may mostly be interested in **Phone** product and **Apple** brand. Under this circumstance, previous methods without introducing auxiliary time information will regard it as a session with evenly and moderate length time intervals. Therefore, a deviation from the major interests may be caused by the latest items, which are laptops. While for the second session with a relatively long time interval between *iPhone* and *Macbook*, we argue that the interest drift may occurred during the interval. Thus the interests distributed in the front portion of timeline such as **Phone**, ought to be a minor factor for recommendation, while the interest **Laptop** in the latter portion should be taken into special consideration. But for SBR methods ignoring time information, the multiple interests in two chunks of the session cannot be effectively disentangled and rearranged. The minor interest **Phone** is likely to be over weighted for user's current intention, while the crucial interest **Laptop** may be diluted. As discussed above, even sessions with exactly the same items and orders, but for diverse time distributions, may have diverse interests intensity, leading to completely different next items. In this case, the ignorance of time information makes existing methods insufficient to distill effective intention signals from the active session.

To address these problems, we propose *Temporal aware Multi-Interest Graph Neural Network* (TMI-GNN), a novel method to distill the disentangled interest and inject the temporal information for better inferring the user intentions of the current session. To be more specific, we firstly construct a multi-interest graph for current session by appending multiple interest nodes into original item-item graphs, which builds the potential relations between items and different interest factors. Then, to generate the item and interest representations, we synchronously apply item information propagation with item-level time interval signals, interest extraction in a soft clustering way, and interest attaching with the interest-level distribution information. Finally, we integrate item embeddings and interest embedding under the guide of temporal distance, to represent user's preference for next item prediction.

Our main contributions of this work are summarized below:

- We propose to construct a multi-interest graph with item and interest nodes for representing multi-interest session effectively, involving the explicit item transitions and potential connections between different items and interests.

- For the constructed graph, we develop a novel **TMI-GNN** model for SBR to capture adjacent item transitions, distill users' current multi-interests from noisy interaction sequences and feedback related interest information to enhance session representations. Moreover, it explicitly injects item-level and interest-level temporal information into the above process to refine current intention representation.

- Extensive experiments on three datasets demonstrate that our model is superior compared with state-of-the-art models.

2. Related Works

Session-based Recommendation. Following the development of deep learning, many neural network based approaches have been proposed for SBR. Due to well sequence modeling capability of RNNs, RNN-based methods have been widely used for SBR (Hidasi et al. (2015); Li et al. (2017); Liu et al. (2018); Ren et al. (2019); Song et al. (2019)). For instance, GRU4Rec Hidasi et al. (2015) was firstly proposed to utilize GRU layer to capture information in interaction sequences. Based on GRU4Rec, NARM Li et al. (2017) added an attention mechanism after RNN, which refers to last interaction item, in order to capture the global and local preference representation of user in current session. But in an ongoing session, interaction patterns are always more complex than simple sequential signals, which cannot be effectively captured by RNN-based models. More recently, motivated by the superior performance of GNNs in extracting complex relationships between objects, quite a few recommendation methods relying on GNNs were proposed to extract the item transition patterns for SBR (Wu et al. (2019); Xu et al. (2019); Qiu et al. (2019); Pan et al. (2020); Chen and Wong (2020); Wang et al. (2020b); Pang et al. (2021); Chen et al. (2021a)). For example, SRGNN Wu et al. (2019) converted the interaction sequence into a directed graph and employed the gated GNN (GGNN) on the graph to learn item embedding. LESSR Chen and Wong (2020) formed better graph structure from the session by proposing a lossless encoding scheme, and proposed a SGAT layer to model the long-range dependency, which propagates information along shortcut graph. StarGNN Pan et al. (2020) put forward a star graph neural network to model the complex transitions between items with additional node to connect the long-range item relations. DATMDI Chen et al. (2021a) learned the cross-session and local session representations via the GNN and GRU and combined them by a soft-attention mechanism.

Temporal information in Recommendation. Meanwhile, temporal information is also a key feature in many recommendation scenarios, such as e-commerce and video recommendation. There are a few works to utilize the temporal information as contextual feature for recommendation (Vassøy et al. (2019); Li et al. (2020); Wu et al. (2017); Gu et al. (2020)). However, most existing methods simply reduce the temporal information to order relationship, subsequently use RNN-based model to capture the sequential signal. For instance, Yu et al. (2016) used RNN to learn a dynamic representation of a user which reveal user's dynamic interests at different time in next basket recommendation. Ye et al. (2020) discovered absolute time patterns and relative time patterns based on insightful data analysis to model users' temporal behaviors for recommender systems. Zhou et al. (2018b) proposed a framework called RIB, which takes dwell time into account.By using the time a user spends on one item as a part of so called micro behavior, it can model more practical user intent.

As outlined above, previous works on SBR have some limitations. First, temporal information is rarely or crudely exploited in these works. What's more, none of these methods explicitly disentangle the multiple interests of user in a session. These limitations may lead to suboptimal performance.

3. Preliminary

In this work, we aim to explore the effectiveness of abundant temporal information and multiple disentangled interest for SBR. Given the entire item set $\mathcal{V}$, we first define a timestamp-
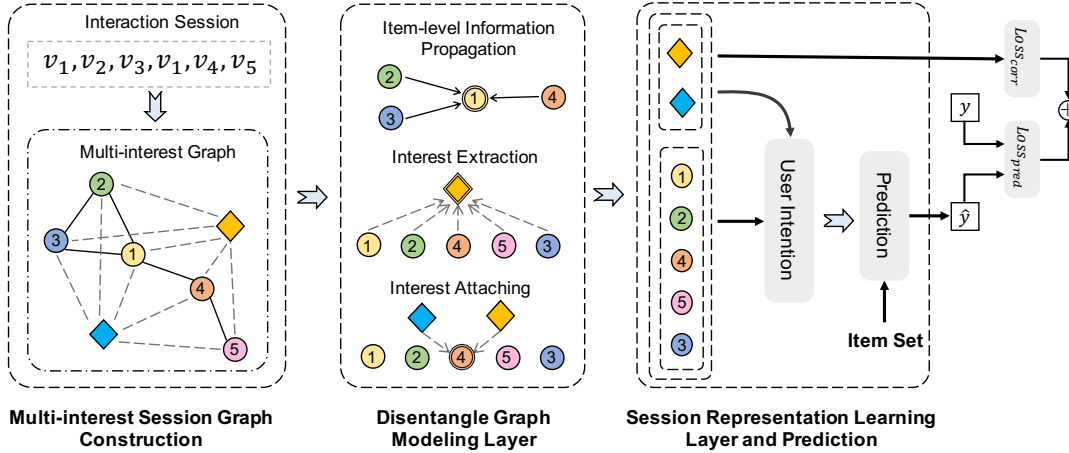
Figure 2: The overview of TMI-GNN.

augmented item session sequence as $s = \{(v_1, t_1), (v_2, t_2), ..., (v_L, t_L)\}$, where (v_i, t_i) represents the user interacted item $v_i \in \mathcal{V}$ at time t_i and $t_i < t_{i+1}$, and L is the session length. Given session s, the goal of SBR is to predict a probability score for any item $v \in \mathcal{V}$ such that an item with higher score is more likely to be interacted next.

4. Methodology

4.1. Overview

In this section, we detail the design of our model. As shown in Figure 2, it contains four main components: Multi-interest Session Graph Construction, Disentangle Graph Modeling Layer, Session Representation Learning Layer and Prediction Layer. By constructing item-transition sequences as multi-interest graphs with additional interest nodes, we explicitly build the potential connection between latent user preference and explicit items sequence. Furthermore, at the Disentangle Graph Modeling Layer, we extract item transition information, distill multi-interest representations and feedback disentangled interests to related item embeddings, respectively. With the guidance of item-level and interest-level temporal information, the refined attention is better estimated for disentangling user's multi-interest. Then, Session Representation Learning Layer adaptively generates the final intention representations with the injection of last-item based time interval information. Finally, the predictor estimates the probability of candidate items based on the each disentangled session representations.

4.2. Multi-interest Session Graph Construction

As mentioned above, the extraction of multi-interest is meaningful for obtaining user intention representation from the interaction sequence. Therefore, we propose multi-interest session graph for effectively organizing the interactions in the session and modeling user's multi-interest.

For given session s, we construct the multi-interest session graph $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$,where $\mathcal{V}_s = (\{v_1, v_2, v_3, \ldots, v_N\}, \{u_1, u_2, \ldots, u_H\})$ indicates the node set of the constructed graph which contains N item nodes and H interest nodes. Besides the basic transition between

items in the session, we additionally introduce the **interest nodes** to represent each independent interest, which can be explained as different distributions of items' contribution to user's intention. For j-th interest node, it fully connects to all items in the session with edges $\{(u_j, v_i) | 1 \leq i \leq N\}$. For each session, each item node v_i has corresponding item-interest edges $\{(v_i, u_j) | 1 \leq j \leq H\}$ to each interest node, and item-item transition edge to contextual item nodes. Through the full connection between the explicit item and interest node, the soft assignment of each item to corresponding interest can be estimated by subsequent GNN as edge attribute.

4.2.1. Temporal Information.

Furthermore, we attach auxiliary multi-form temporal information to the multi-interest graph for distilling more precise interest representations. For the original timestamp sequence $(t_1, t_2, \ldots, t_L)$ of session s, item-level transition interval $t_{i,j}$ is attached to the edge of item i and j:

$$t_{i,j} = \frac{|t_i - t_j|}{t_{bucket}} \, , \tag{1}$$

where t_{bucket} denotes the pre-defined length of time bucket. Moreover, the relative time-step $t_{i,1}$ is attached to the interaction position i, compared to the start time at the first position.

For the constructed multi-interest session graph, there are three types of relation, i.e., item-item, item-interest and interest-item relation, and we represent them by superscript $v \to v$, $v \to u$ and $u \to v$ respectively.

4.3. Disentangle Graph Modeling Layer

Next, we present how to obtain item representations and disentangle interest representations on the constructed heterogeneous multi-interest graph. Based on the message propagation of GNNs, the item and interest node embeddings are updated based on the previous results with neighbor information. The disentanglement of multi-interest can be considered as the process of iteratively refining interest node embedding, and the explainable assignments for each interest factor can be estimated as the weight of item-interest edges based on the full connection of each independent interest node and the item nodes, i.e., each interest factor is generated by item information pooling with different assignment scores. For the adjacent item transition, previous GNN-based model, like SRGNN (Wu et al. (2019)), has achieved superior performance for SBR. Therefore, for each GNN layer, we firstly aggregate the neighbor messages in each relation respectively. Then we gather the semantic information to update the item and interest representations.

Let $\mathbf{u}_i^{(k)}, \mathbf{v}_j^{(k)}$ denote the embedding of interest u_i and item v_j after k layers GNN propagation. The item IDs are embedded into d-dimensional space and are used as initial node features in our model, $\mathbf{v}_j^{(0)} \in \mathbb{R}^d$. For the multiple temporal information, we embed them by a learnable temporal matrix $\mathbf{T} = [\mathbf{T}_0, \mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_m]$ for the rounded time value $0 \leq t \leq m$, where m is the max time-step intervals.

Moreover, for each interest factor as related interest node, we adopt average operation on the item nodes to initialize the interest representation, i.e., $\mathbf{u}_i^{(0)} = \frac{1}{L} \sum_{j=1}^L \mathbf{v}_j^{(0)}$. For the interest-side temporal information, we utilize the the center timestamp $t_{cent,i}$ to indicate the location of i-th interest factor on the timeline, and the temporal compactness value $t_{comp,i}$

to indicate the coverage of interest factors in the item sequence. Here we initialize these two characteristics of each interest factor with average pooing, i.e., $t_{cent,i}^{(0)} = \frac{1}{L}\sum_{j=1}^{L} t_{j,1}$, and $t_{comp,i}^{(0)} = \frac{1}{L}\sum_{j=1}^{L} |t_{j,1} - t_{cent,i}^{(k)}|$.

Then, we will detail the modeling for item-item, item-interest and interest-item relations, respectively.

4.3.1. ITEM-LEVEL INFORMATION PROPAGATION LAYER

For the item-item relation, we adopt SRGNN with detail time interval information to propagate adjacent node information. It assembles neighbor node information with the normalized coefficient $e_{ij}^{v \rightarrow v}$ under the guide of time interval signal t_{ij}:

$$e_{ij}^{v \rightarrow v} = \operatorname*{softmax}_{j \in \mathcal{N}_{v \rightarrow v}(i)} \left(\mathrm{MLP} \left(\mathbf{T}_{t_{i,j}} \right) \right) , \tag{2}$$

$$\mathbf{m}_i^{(k)} = \sum_{j \in \mathcal{N}_{v \rightarrow v}(i)} e_{ij}^{v \rightarrow v} \mathbf{v}_j^{(k)} , \tag{3}$$

where MLP represents a simple multilayer perceptron for time interval embedding $\mathbf{T}_{t_{i,j}}$, and shares with different GNN layers. Similar to GGNN, we feed the neighbor information $\mathbf{m}_i^{(k)}$ and the previous layer item $\mathbf{v}_i^{(k)}$ into GRU to update the item-item relation node representations:

$$\mathbf{v}_i^{v \rightarrow v,(k+1)} = \mathrm{GRU} \left(\mathbf{v}_i^{(k)}, \mathbf{m}_i^{(k)} \right) , \tag{4}$$

where the GRU $unit$ is parameters-shared for all item nodes updating at current layer. With the combination of contextual interaction items representations and previous cross-semantic item presentations, we integrate the chronological item transition information into node embedding in the item-item relation.

4.3.2. INTEREST EXTRACTION LAYER.

For the item-interest relation at layer k, a graph attention neural network is utilized for updating interest node embedding, as the process of distilling interest representation. In particular, we compute the corresponding correlation weight $\alpha_{ij}^{(k)}$ between target interest node u_i and neighbor item v_j, as the explainable assignment scores for disentangled interest. And interest node representation is updated via the sum pooling as following:

$$\alpha_{ij}^{(k)} = \operatorname*{softmax}_{j \in \mathcal{N}_{v \rightarrow u}(i)} \left(\mathrm{LeakyReLU} \left(\mathbf{W}_u^{v \rightarrow u,(k)} \mathbf{u}_i^{(k)} + \mathbf{W}_v^{v \rightarrow u,(k)} \mathbf{v}_j^{(k)} \right) \right) , \tag{5}$$

$$u_i^{v \rightarrow u,(k+1)} = \sum_{j \in \mathcal{N}_{v \rightarrow u}(i)} \alpha_{ij}^{(k)} \mathbf{W}_{trans}^{v \rightarrow u,(k)} \mathbf{v}_j^{(k)} . \tag{6}$$

where $\mathbf{W}_{trans}^{v \rightarrow u,(k)} \in \mathbb{R}^{d \times d}$ represents the information transformation matrix of neighbor nodes, $\mathbf{W}_u^{v \rightarrow u,(k)}, \mathbf{W}_v^{v \rightarrow u,(k)} \in \mathbb{R}^{1 \times d}$ are the parameters of linear transformation for the target interest and the source item, respectively.

Meanwhile, we also update the center timestamp and temporal compactness of each interest node based on the normalized correlation coefficient:

$$t_{cent,i}^{(k+1)} = \sum_{j \in \mathcal{N}_{v \rightarrow u}(i)} \alpha_{ij}^{(k)} t_{j,1} \ ; \ t_{comp,i}^{(k+1)} = \sum_{j \in \mathcal{N}_{v \rightarrow u}(i)} \alpha_{ij}^{k} |t_{j,1} - t_{cent,i}^{(k)}|. \tag{7}$$

Through the assignment score $\alpha_{ij}^{(k)}$, we not only distill the latent representations of multi-view interests from the sessions with overlapped and interwove interests, but also offer an explanations parameter for each interest factor.

4.3.3. Interest Attaching Layer

To refine the item representation via the disentangled interest, we adopt a graph attention network to update the item representation under the interest-item semantic. At first, we estimate the attention coefficient between target item and source interest node. Here, we consider both the similarity and temporal continuity of each item and interest node pair:

$$\beta_{ij}^{(k)} = \operatorname*{softmax}_{j \in \mathcal{N}_{u \to v}(i)} \left(\sigma \left(\left(\mathbf{W}_v^{u \to v,(k)} \mathbf{v}_i^{(k)} \right)^T \mathbf{W}_u^{u \to v,(k)} \mathbf{u}_j^{(k)} + \mathbf{w}_t^T \mathbf{T}_{t_{i,j}^{u \to v,(k)}} + b_t \right) \right), \quad (8)$$

where $\mathbf{W}_v^{u \to v,(k)}, \mathbf{W}_u^{u \to v,(k)} \in \mathbb{R}^{d \times d}$ are the transforming matrices for the item and interest, σ is an activate function. $t_{i,j}^{u \to v,(k)} = \frac{\left| t_{cent,j}^{(k)} - t_{i,1} \right|}{t_{comp,j}^{(k)}}$ denotes the distance between item and interest in timeline, and $\mathbf{w}_t \in \mathbb{R}^d, b_t$ are the linear transformation of it. Here, we utilize the residual between the center timestep of interest and timestep of items regularized by the interest compactness value, to measure the similarity of interest and item pair in the temporal view. Then, gathered interest factors are scattered to item nodes for generating the intention-augmented item representations via parameters $\mathbf{W}_{trans}^{u \to v,(k)} \in \mathbb{R}^{d \times d}$:

$$\mathbf{v}_i^{u \to v,(k+1)} = \sum_{j \in \mathcal{N}_{u \to v}(i)} \beta_{ij}^{(k)} \mathbf{W}_{trans}^{u \to v,(k)} \mathbf{u}_j^{(k)}, \quad (9)$$

The edges between interests and items indirectly connect each item to all other items through the intermediate interest node. Compare to the original sparse item-item graph, the additional interest nodes and edges help GNNs to effectively capture long-range dependencies in sessions with any length because it propagates information along intermediate interest paths within two-hop.

4.3.4. Layer Combination

By updating the node representation based on diverse semantic relations synchronously, we obtain the item representations from adjacent items and related interest, and the interest embeddings based on item-level soft clustering. Then, we gather cross-semantic information through average operation like RGCN (Schlichtkrull et al. (2018)), i.e., $\mathbf{v}_j^{(k+1)} = \sigma(\mathbf{v}_j^{u \to v,(k+1)} + \mathbf{v}_j^{v \to v,(k+1)})/2$, $\mathbf{u}_i^{(k+1)} = \sigma(\mathbf{u}_i^{v \to u,(k+1)})$, and consider $\mathbf{v}_j^{(k+1)}$ and $\mathbf{u}_i^{(k+1)}$ as the output of item node v_j and interest node u_i after k-th GNN layers. Moreover, in order to mine deeper items transition relations, multi-layers of GNN are stacked to propagate high-order information. And we utilize the gated mechanism to balance the item node representations between initial embedding and K layers output, as follows:

$$g = \sigma(\mathbf{W}_{gated} \left[\mathbf{v}_j^{(0)} \parallel \mathbf{v}_j^{(K+1)} \right]), \quad (10)$$

$$\mathbf{v}_j = g\mathbf{v}_j^{(0)} + (1-g)\mathbf{v}_j^{(K+1)}. \quad (11)$$

where $\parallel$ is the concatenate operation, $\mathbf{W}_{gated} \in \mathbb{R}^{2d}$ and sigmoid function $\sigma(\cdot)$ generate the balance factor g to alleviate over-smooth problems of deep GNN. As for the interest

node, we simply adopt the multi-layers' output as the final latent interest representation, i.e., $\mathbf{u}_i = \mathbf{u}_i^{(K+1)}$.

4.4. Session Representation Learning Layer

After the Disentangle Graph Modeling, we obtain the final item and interest node embeddings. Then we aggregate the representation of items in the session based on each abstract interest factor to generate the session representations. Moreover, the items clicked later in the session usually reveal the user's current intentions better, which draws greater attention for SBR. Therefore, we incorporate additional last-item based time interval information into interest-based attention to capture user activate intention dynamically.

In detail, inspired by the reverse position embedding in GCE-GNN (Wang et al. (2020b)), we integrated the last-item based time interval information with the obtained item representations, which extends the position information to a more fine-grained time domain to make a better prediction:

$$\mathbf{z}_i = \tanh\left(\mathbf{W}_0\left[\mathbf{v}_i \parallel \mathbf{T}_{t_{L,i}}\right] + \mathbf{b}_0\right), \tag{12}$$

where $\mathbf{W}_0 \in \mathbb{R}^{d \times 2d}$ and $\mathbf{b}_0 \in \mathbb{R}^d$ are trainable parameters, $\mathbf{T}_{t_{L,i}}$ is the embedding for last-time based interval $t_{L,i}$. Then, the intentions of user are learnt by a shared attention mechanism, which dynamically weights item representation based on each interest $\mathbf{u}_h$:

$$\gamma_{i,h} = \mathbf{q}^T \sigma(\mathbf{W}_1 \mathbf{z}_i + \mathbf{W}_2 \mathbf{u}_h + \mathbf{b}), \quad \mathbf{s}_{G,h} = \sum_{i=1}^{N} \gamma_{i,h} \mathbf{v}_i, \tag{13}$$

where σ is an activate function, $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d \times d}$ and $\mathbf{q}, \mathbf{b} \in \mathbb{R}^d$ are trainable parameters. Then we combine the h-th coarse-level intention and refined item-augment representations to generate the session representation for each interest:

$$\mathbf{S}_h = \sigma\left(\mathbf{W}_3\left[\mathbf{s}_{G,h} \parallel \mathbf{u}_h\right]\right), \tag{14}$$

where $\mathbf{W}_3 \in \mathbb{R}^{d \times 2d}$ is the trainable parameter.

With the incorporation of auxiliary last-item based interval information, we capture session representations involved in both the disentangled interest factors and interaction session items in relative chronological temporal-aware patterns.

4.5. Prediction and Training

Based on each disentangled interest representation $\mathbf{S}_h$ learnt above and the normalized initial embeddings $\mathbf{v}_i'$ of candidate items, we then estimate the interaction probability $\hat{\mathbf{y}}$ of candidate items for current session:

$$\hat{\mathbf{y}}_i = \max_{1 \leq h \leq H} \text{softmax}\left(\mathbf{S}_h^T \mathbf{v}_i'\right), \quad \mathbf{v}_i' = \mathbf{v}_i^{(0)} / \parallel \mathbf{v}_i^{(0)} \parallel_2 \tag{15}$$

where $\hat{y}_i \in \hat{\mathbf{y}}$ denotes the probability that the user will click on item v_i in the current session, and H is the pre-defined parameter of the interest node number.

As mentioned above, flexible number of interest nodes encourages the chunked interest representations conditioned on different behavior patterns. However, the difference constraint drove by multiple interest extractions is insufficient: there might be redundancy among latent interests representation, which conflicts with the target of disentangling multi-view user interest. We hence introduce interest independents loss, which hires distance

correlation measures as a regularizer, with the target of encouraging the multi-interest representations to be diverse. We formulate this as follows:

$$\mathcal{L}_{corr} = \sum_{i=1}^{H} \sum_{j=i+1}^{H} \cos\left(\mathbf{u}_i, \mathbf{u}_j\right) , \tag{16}$$

where $cos\left(\cdot\right)$ indicates the similarity distance between two inner-session interest representation pair. The final optimization process is to minimize the cross entropy loss function together and the interest-independence loss jointly:

$$\mathcal{L} = -\sum_{i=1}^{|\mathcal{V}|} \mathbf{y}_i \log(\hat{\mathbf{y}}_i) + (1 - \mathbf{y}_i)\log(1 - \hat{\mathbf{y}}_i) + \lambda \mathcal{L}_{corr} , \tag{17}$$

where $\mathbf{y}_i \in \mathbb{R}^{|\mathcal{V}|}$ is a one-hot vector of ground truth, and λ is the coefficient controlling interest-independence term.

5. Experiments

In this section, we conduct experiments on SBR to evaluate the performance of our method compared with other state-of-the-art models. Our purpose is to answer the following research questions: **RQ1:** How does our model perform compared with state-of-the-art SBR methods? **RQ2:** How does the temporal information of the sequence affect the recommendation results? **RQ3:** Is the design of our model reasonable and effective? How do the key modules of TMI-GNN influence the model performance? **RQ4:** How do the hyper-parameters affect the effectiveness of our model?

5.1. Experimental Setup

5.1.1. Dataset.

We conduct extensive experiments on three public datasets: *RetailRocket*, *Yoochoose* and *Jdata*, which are widely used in the SBR research (Chen and Wong (2020); Wang et al. (2020b); Meng et al. (2020)) and can support our work with seconds-level timestamp information. *RetailRocket* contains behavior data and item properties that collected from a real-world e-commerce website. *Yoochoose* is a RecSys Challenge dataset, which consists of click-streams from an E-commerce website. Different from Liu et al. (2018); Li et al. (2017), we use the most recent fractions 1/16 sequences of Yoochoose as the total dataset *Yoochoose* 1/16. *Jdata* records the historical interactions from the JD.com. It contains a stream of user actions within two month. We extract the session data with the setting of the duration time threshold 1 hour.

To filter poorly informative sessions and items, following Li et al. (2017); Wu et al. (2019), we first filtered out all sessions of length ≤ 2 and items appearing less than 5 times in all datasets. Then we applied a data augmentation technique described in Li et al. (2017). The statistics of all datasets after prepossessing are summarized in Table 1.

5.1.2. Baseline Models.

To demonstrate TMI-GNN's superiority performance, we compare it with several representative competitors, including the state-of-the-art SBR models and several temporal-

Table 1: Statistics of datasets used in experiments.

Statistic	RetailRocket	Yoochoose	Jdata
No. of items	45,630	19,589	92,792
No. of sessions	341,831	975,060	3,397,208
Avg. of session length	4.91	6.87	6.68
Avg. of time interval (seconds)	65.8	51.3	52.6
No. of train sessions	314,924	777,025	2,862,842
No. of test sessions	26,907	198,035	534,366

concerned methods. **(1).GRU4Rec**Hidasi et al. (2015) employs the GRU to capture the representation of the item sequence simply. **(2).NARM** Li et al. (2017) is a RNN-based model which combines with attention mechanism to generate the session embedding. **(3).RIB** Zhou et al. (2018b) is a framework using RNN and attention layer to model user micro behavior including behavior and dwell time. Here we consider the time intervals as the dwell time and ignore the behavior types. **(4).SRGNN** Wu et al. (2019) converts session sequences into directed unweighted graphs and utilizes a GGNN layer Li et al. (2015) to learn the patterns of item transitions. **(5).LESSR** Chen and Wong (2020) adds shortcut connections between items in the session and considers the sequence information in graph convolution by using GRU. **(6).DATMDI** Chen et al. (2021a) combines the GNN and GRU to learn the cross-session enhanced session representation.

Besides, we combine the time interval embdding with ID embedding as input following Zhou et al. (2018b), and inject additional time information into the graph by adding learnable time interval weights like section 4.3.1 for SRGNN, LESSR and DATMDI method separately, named SRGNN†, LESSR†, DATMDI†.

5.1.3. Evaluation Metrics.

To evaluate the recommendation performance, we employ two widely used metrics: Hit ratio (H@k) and Normalized discounted cumulative gain (N@k) following Wu et al. (2019), where k is 10 or 20. The average results over all test session are reported.

5.1.4. Implementation Details.

We implement the proposed model based on Pytorch and DGL. The embedding dimension is set to 128. All parameters are initialized through a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. We employ the Adam optimizer to train the models with the mini-batch size of 512. We conduct the grid search over hyper-parameters as follows: learning rate in $\{0.001, 0.01, 0.1\}$, learning rate decay in $\{0.01, 0.05, 0.1, 0.5\}$, learning rate decay step in $\{2, 3, 4\}$, controlling factor λ in $\{1, 3, 10, 30\}$. The maximal time-step m is set to 300, which is large enough for all sessions. To make the comparison fairer, we range the hyper-parameters of baseline methods with the same tuning scopes of our experiments.

Table 2: Experimental results (%) of different models in H@{10, 20}, and N@{10, 20} on three datasets. *Improv.* means improvement over the state-of-art methods.The bold number indicates the improvements over the best baseline (underlined) are statistically significant ($p<0.01$) with paired t-tests.

Models	RetailRocket				Yoochoose 1/16				Jdata			
	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20
GRU4Rec	29.40	37.32	18.64	20.73	44.91	54.12	28.16	30.29	28.62	39.05	15.67	18.58
NARM	32.16	41.09	20.23	21.89	52.86	63.12	32.03	34.64	31.47	43.32	16.41	19.40
SRGNN	32.32	40.60	20.13	22.10	54.87	64.63	33.90	36.38	33.01	44.27	18.88	21.73
LESSR	31.82	40.48	19.90	22.09	55.51	65.28	34.51	36.99	33.13	44.01	18.99	21.74
DATMDI	32.26	40.51	20.06	22.15	55.77	65.58	34.61	37.12	33.24	44.39	19.05	21.82
RIB	32.36	40.96	20.17	21.78	52.80	63.08	31.97	34.59	31.36	43.24	16.31	19.33
SRGNN†	32.41	40.63	20.11	22.12	54.85	64.64	33.87	36.39	33.10	44.37	18.93	21.79
LESSR†	31.90	40.56	19.94	22.15	55.59	65.35	34.60	37.09	33.23	44.12	19.03	21.76
DATMDI†	32.46	40.64	20.12	22.13	55.86	65.62	34.67	37.25	33.29	44.46	19.09	21.88
Ours	**33.20**	**42.68**	**21.04**	**22.97**	**56.81**	**66.58**	**35.72**	**38.31**	**33.94**	**45.43**	**19.49**	**22.35**
Improv.	2.23%	3.72%	3.84%	1.22%	1.70%	1.46%	2.97%	2.84%	1.95%	2.18%	2.09%	2.14%

5.2. Overall Comparison (RQ1)

To demonstrate the overall performance of the proposed model, we compared it with the state-of-the-art methods for SBR. We can obtain the following significant observations from the comparison results shown in Table 2.

Comparison of Different Baselines. The NARM performs significantly better than GRU4Rec, which indicates the effectiveness of attention mechanism to capture the user's main motivation. In the comparison between RNN-based models and GNN-based models for *Yoochoose* and *Jdata*, the GNN-based models generally outperform the RNN-based models, which verified the certain advantage of GNN for SBR. Moreover, we notice that LESSR can outperform SRGNN in most cases. This may be due to the special design in LESSR for capturing long-range dependency between items in a session, which contributes to better performance on some long sessions. Meanwhile, for dataset *RetailRocket*, the simple model NARM outperforms other complex models because of the insufficient training caused by the small amount of data.

Significance of Temporal Information. Then we turn to the temporal information attached methods. The RNN-based model RIB achieves superior performance to GRU4Rec but performs slightly inferior than NARM, because RIB employs a relatively simple attention instead of last-item based attention. Compared to the original models, the temporal-enhanced methods all achieve better performance in some degrees, which indicates the significance of temporal information in SBR. Moreover, the adapted methods perform better on small dataset RetailRocket, which indicates that the auxiliary temporal information is more helpful for sparse interaction data.

Model Effectiveness. We find that our model comprehensively outperforms all other baselines substantially on almost all metrics, which justifies the effectiveness of our model. The performance improvement can be explained in two aspects. One is that TMI-GNN can disentangle user intention via extracting multi-interests from the multi-interests session

graph, so it can portray more profound representation of user interests. This strategy break the limitation of expression ability of only one interest. Another one is that we introduce multi-form temporal information to the process of disentangle graph modeling and session representation learning. Compare with other time-aware models, TMI-GNN models diversified temporal information adequately effectively.

Table 3: The performance comparison w.r.t different temporal information and module design.

Model setting		-V2V	-U2V	-Last	First	-Interest	-Loss	Ours
Yoochoose	H@20	66.32	66.25	66.28	66.37	65.86	66.35	66.58
	N@20	37.54	37.63	37.53	37.79	37.23	37.71	38.31
Jdata	H@20	44.89	44.75	44.82	45.14	44.25	45.21	45.43
	N@20	21.98	21.94	21.97	22.02	21.80	22.03	22.35

5.3. Ablation Study (RQ2&RQ3)

Impact of temporal information. In this part, we compare our model with partially temporal information masked versions in Table 3 to test whether considering the multi-form temporal information can boost model performance. The method with "-V2V" means skipping the time intervals in item-level message propagation, "-U2V" indicates ignoring the temporal factors in interest-item relation, "-Last" represents removing the last-item based time-interval signals, and "First" means utilizing the first-item based time-interval embedding in session representation learning module. By comparing methods mentioned above, we find that the loss of any type of temporal information will cause the decline of model performance. Besides, compare to "First" time embedding, the last-item based time-interval information is more helpful for SBR. Moreover, the loss of interest-item temporal continuity factors is more significant in our model compared with other temporal information types. Based on the above illustrations, we demonstrate that injecting multi-form temporal information in our framework is indeed meaningful.

Impact of different Designs. In this part, we compare our method with different variants to verify the effectiveness of the critical components of TMI-GNN. Specifically, we remove the additional interest node in session graph (denote as "-Interest"), and mask the interest independent loss (denote as "-Loss"), respectively. The experimental results are presented in Table 3. It can be observed that the abstract interest nodes is pivotal for the model capability of intention capturing. Meanwhile, for the diverse intention modeling, the removal of interest independent loss leads to great impact on model results, which demonstrates that the forced cross-interest separation is helpful for disentangling multiple interest of user. In summary, we can infer that the key components of TMI-GNN are effective through the comparison and analysis above.

5.4. Hyper-parameters Study (RQ4)

Impact of time bucket widths. As discussed in section 4, we divide the time interval into buckets to utilize time signals. So we conduct tests by ranging the time bucket width within $\{2, 4, 8, 16, 32\}$ to explore how does the bucketing setting affect the model's performance. As shown in Figure 3, we can see that the performance does not fluctuate dramatically as
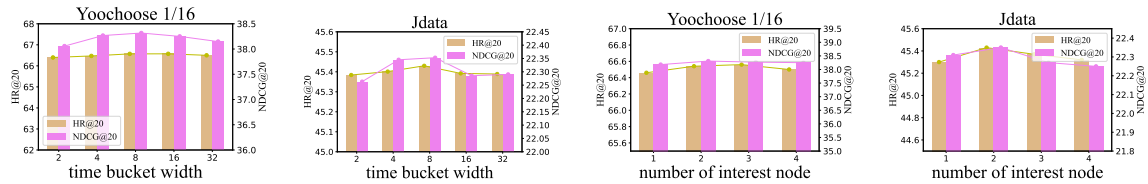
SHEN♠ ZHU♠ PANG ZHANG WEI

Figure 3: Performance comparison w.r.t. different time bucket width and max interest node number.

the time bucket width changes, which indicates that our model is not sensitive to the time bucket width.

Impact of interest node num. To investigate the impact of the interest node number, we range this parameter in $\{1, 2, 3, 4\}$. According to the results in Figure 3, we can see that for both *Yoochoose* and *Jdata*, the model with 2 interest nodes achieves the best performance in most metrics. Compare to single interest node with mixed interest representation, our model with two interest nodes disentangles the latent interest representation for better next item prediction. When the number becomes larger, performance will drop due to the redundancy of interest representation.

6. Conclusion

In this paper, we pay special attention to the disentangled multi-interest representations of user and multiple temporal information for session based recommendation. We construct a multi-interest graph and devise the TMI-GNN model, which utilizes the multi-interest graph to capture adjacent item transitions, distill multi-interest representations with the injection of multi-form temporal information. In the experiments, our model outperforms other state-of-the-art session-based models, showing the effectiveness of our model.

Acknowledgments

References

Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. Controllable multi-interest framework for recommendation. In *SIGKDD*, 2020.

Chen Chen, Jie Guo, and Bin Song. Dual attention transfer in session-based recommendation with multi-dimensional integration. In *SIGIR*, SIGIR '21, page 869–878, 2021a.

Tianwen Chen and Raymond Chi-Wing Wong. Handling information loss of graph neural networks for session-based recommendation. In *SIGKDD*, pages 1172–1180, 2020.

Wanyu Chen, Pengjie Ren, Fei Cai, Fei Sun, and Maarten De Rijke. Multi-interest diversification for end-to-end sequential recommendation. *ACM Trans. Inf. Syst.*, 2021b.

Yulong Gu, Zhuoye Ding, Shuaiqiang Wang, and Dawei Yin. Hierarchical user profiling for e-commerce recommender systems. In *WSDM*, pages 223–231, 2020.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *WWW*, pages 173–182, 2017.

Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *TOIS*, 2004.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. Multi-interest network with dynamic routing for recommendation at tmall. In *CIKM*, pages 2615–2623, 2019.

Jiacheng Li, Yujie Wang, and Julian McAuley. Time interval aware self-attention for sequential recommendation. In *WSDM*, pages 322–330, 2020.

Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. Neural attentive session-based recommendation. In *CIKM*, pages 1419–1428, 2017.

Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. Stamp: short-term attention/memory priority model for session-based recommendation. In *SIGKDD*, 2018.

Yujie Lu, Shengyu Zhang, Yingxuan Huang, Luyao Wang, Xinyao Yu, Zhou Zhao, and Fei Wu. Future-aware diverse trends framework for recommendation. In *WWW*, 2021.

Wenjing Meng, Deqing Yang, and Yanghua Xiao. Incorporating user micro-behaviors and item knowledge into multi-task learning for session-based recommendation. In *SIGIR*, pages 1091–1100, 2020.

Zhiqiang Pan, Fei Cai, Wanyu Chen, Honghui Chen, and Maarten de Rijke. Star graph neural networks for session-based recommendation. In *CIKM*, pages 1195–1204, 2020.

Yitong Pang, Lingfei Wu, Qi Shen, Yiming Zhang, Zhihua Wei, Fangli Xu, Ethan Chang, and Bo Long. Heterogeneous global graph neural networks for personalized session-based recommendation. *arXiv preprint arXiv:2107.03813*, 2021.

Ruihong Qiu, Jingjing Li, Zi Huang, and Hongzhi Yin. Rethinking the item order in session-based recommendation with graph neural networks. In *CIKM*, pages 579–588, 2019.

Pengjie Ren, Zhumin Chen, Jing Li, Zhaochun Ren, Jun Ma, and Maarten De Rijke. Repeat-net: A repeat aware neural recommendation machine for session-based recommendation. In *AAAI*, volume 33, pages 4806–4813, 2019.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer, 2018.

Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. Autorec: Autoencoders meet collaborative filtering. In *WWW*, pages 111–112, 2015.

Jing Song, Hong Shen, Zijing Ou, Junyi Zhang, Teng Xiao, and Shangsong Liang. Islf: Interest shift and latent factors combination model for session-based recommendation. In *IJCAI*, pages 5765–5771, 2019.

Bjørnar Vassøy, Massimiliano Ruocco, Eliezer de Souza da Silva, and Erlend Aune. Time is of the essence: a joint hierarchical rnn and point process model for time and item predictions. In *WSDM*, pages 591–599, 2019.

Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. Disentangled graph collaborative filtering. In *SIGIR*, pages 1001–1010, 2020a.

Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. Global context enhanced graph neural networks for session-based recommendation. In *SIGIR*, 2020b.

Le Wu, Yong Ge, Qi Liu, Enhong Chen, Richang Hong, Junping Du, and Meng Wang. Modeling the evolution of users' preferences and social links in social networking services. *TKDE*, pages 1240–1253, 2017.

Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. Session-based recommendation with graph neural networks. In *AAAI*, volume 33, pages 346–353, 2019.

Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. Graph contextualized self-attention network for session-based recommendation. In *IJCAI*, volume 19, pages 3940–3946, 2019.

Wenwen Ye, Shuaiqiang Wang, Xu Chen, Zheng Qin, and Dawei Yin. Time matters: Sequential recommendation with complex temporal information. In *SIGIR*, 2020.

Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. A dynamic recurrent model for next basket recommendation. In *SIGIR*, page 729–732, 2016.

Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *SIGKDD*, pages 1059–1068, 2018a.

Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin. Micro behaviors: A new perspective in e-commerce recommender systems. In *WSDM*, pages 727–735, 2018b.