Multi-Faceted Global Item Relation Learning for Session-Based Recommendation

Qilong Han
Harbin Engineering University
Harbin, China
hanqilong@hrbeu.edu.cn

Chi Zhang
Harbin Engineering University
Harbin, China
zhangchi20@hrbeu.edu.cn

Rui Chen*
Harbin Engineering University
Harbin, China
ruichen@hrbeu.edu.cn

Riwei Lai
Harbin Engineering University
Harbin, China
lai@hrbeu.edu.cn

Hongtao Song
Harbin Engineering University
Harbin, China
songhongtao@hrbeu.edu.cn

Li Li
University of Delaware
Newark, United States
lilee@udel.edu

ABSTRACT

As an emerging paradigm, session-based recommendation is aimed at recommending the next item based on a set of anonymous sessions. Effectively representing a session that is normally a short interaction sequence renders a major technical challenge. In view of the limitations of pioneering studies that explore collaborative information from other sessions, in this paper we propose a new direction to enhance session representations by learning multi-faceted session-independent global item relations. In particular, we identify three types of advantageous global item relations, including negative relations that have not been studied before, and propose different graph construction methods to capture such relations. We then devise a novel multi-faceted global item relation (MGIR) model to encode different relations using different aggregation layers and generate enhanced session representations by fusing positive and negative relations. Our solution is flexible to accommodate new item relations and can easily integrate existing session representation learning methods to generate better representations from global relation enhanced session information. Extensive experiments on three benchmark datasets demonstrate the superiority of our model over a large number of state-of-the-art methods. Specifically, we show that learning negative relations is critical for session-based recommendation.

CCS CONCEPTS

• **Information systems → Recommender systems**;

KEYWORDS

Recommender system, session-based recommendation, global item relation

*Corresponding author.

SIGIR '22, July 11–15, 2022, Madrid, Spain

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3532024

ACM Reference Format:
Qilong Han, Chi Zhang, Rui Chen, Riwei Lai, Hongtao Song, and Li Li. 2022. Multi-Faceted Global Item Relation Learning for Session-Based Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3477495.3532024

1 INTRODUCTION

Recommender systems have been widely adopted in various business applications, such as e-commerce and online streaming services, to alleviate information overload, improve user experience, and boost business revenue. Most conventional recommender systems assume that user profiles and long-term historical interactions are available as the basis of recommendation. However, in real-world online services, there are increasing cases where user identities are not revealed (e.g., users who did not log in) or not allowed to be tracked across sessions due to privacy regulations. In these scenarios, only the interactions during an ongoing session are available for recommendation [11, 42], leading to the problem of *session-based recommendation*.

Given a set of *anonymous* sessions, session-based recommendation is aimed at identifying the next item that is most likely to be interacted in the current session. Effectively representing a session lies in the core of session-based recommendation. The majority of existing studies focus on generating better session representations solely based on *individual* sessions. Various deep models have been adopted to learn the short-term intent underlying the current session, including recurrent neural networks (RNNs) [11, 17, 28], convolutional neural networks (CNNs) [33, 50], self-attention mechanisms [19, 43], and graph neural networks (GNNs) [5, 24, 26, 42, 45]. However, sessions are normally of a small number of interactions. Learning from such short interaction sequences *independently* may be inherently inadequate to unveil users' real intents. Consequently, some pioneering studies have started to explore collaborative information from other sessions.

One line of such research proposes to learn from related sessions of a given session, where the relatedness is normally defined by heuristics. For example, Wang *et al.* [39] consider the last m sessions as the related sessions while Luo *et al.* [20] identify related sessions by calculating Jaccard similarity scores among sessions. While these ideas can complement useful collaborative information, they also tend to introduce irrelevant information which could jeopardize

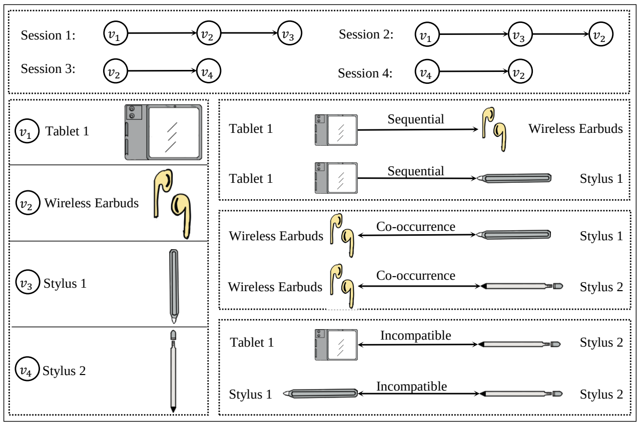


Figure 1: An illustration of multi-faceted global item relations in session-based recommendation.

recommendation performance [41]. Another latest line of research relies on constructing a global graph from all sessions to learn global information, in which items are connected if they are adjacent in some sessions [3, 13, 25, 43, 44, 47]. Compared to the previous line of research, this idea is more systematic. However, it still suffers from a major drawback: such a global graph is built based on only item co-occurrence relations, and thus unable to capture other useful types of global item relations, which are critical to session-based recommendation.

Motivated by the above limitation, in this paper we propose a new direction to enhance session representations by *learning multi-faceted session-independent global item relations*. The key idea is to first derive multi-faceted global graphs, each explicitly modeling a single type of fine-granular global item relations, so that we can learn richer global item embeddings, and then enhance a session's representation using both positive and negative global item embeddings without worrying about measuring the correlations between a given session and other sessions. Our solution reflects a generic framework in that (1) it allows to easily plug in new item relations to further enrich session representations; (2) it can seamlessly integrate different learning methods based on a single session [5, 24, 28, 50, 51] to combine global-level and session-level information. Our design has a few notable advantages. First, learning global item relations in a *session-independent* way is much less computationally expensive, and thus enables to learn more complicated multi-faceted global item relations efficiently. Second, it does not require to leverage rough session representations (e.g., the average of the base item embeddings in a session [41]), which are typically less accurate, to weigh global information. We would like to point out that these advantages well distinguish our solution from GCE-GNN [41] that uses *session-aware* item non-sequential co-occurrence relations to enhance session-level representations.

More specifically, we consider three types of item relations from a global perspective, including *sequential* relations, *non-sequential co-occurrence* relations and *incompatible* relations. Most previous works do not distinguish between sequential and non-sequential co-occurrence relations. However, their distinction is crucial for session-based recommendation [8, 38]. As illustrated in Figure 1, Wireless Earbuds and Stylus 1st Generation are usually purchased after purchasing Tablet 1st Generation in a session, but not in the other way around, suggesting a sequential relation. In contrast, different tablet accessories may not exhibit such sequential relations. Users may buy wireless earbuds and styluses in any order, indicating a non-sequential co-occurrence relation. For simplicity, we use "co-occurrence relation" to mean "non-sequential co-occurrence relation" in the remainder. Both sequential and co-occurrence relations are *positive* relations in the sense that they signify a larger probability of observing a candidate item as the next item given some related items in the current session. In practice, there also exist *negative* relations that help lower the odds of recommending an item. This type of relations has not been previously studied in session-based recommendation or, more broadly, sequential recommendation, which is probably due to its technical challenge. Unlike positive relations that explicitly exist in data, negative relations need to be defined with caution. For example, defining negative relations as two items never co-occurring in any sessions could be misleading when the training data is not large enough.

In this paper, we introduce a type of negative relations called incompatible relations. The general design principle is to resort to more strict observable conditions to capture truly reliable incompatible relations without introducing noisy information. We consider two items as incompatible if they often appear in the same context (i.e., frequently co-exist with certain items in some sessions), but never co-exist in any sessions. Following this data-driven design, we can reveal different useful incompatible relations without requiring any additional prior knowledge (e.g., knowledge graphs). For example, in Figure 1, Tablet 1st Generation and Stylus 2nd Generation are of an incompatible relation because both of them are often purchased with Wireless Earbuds, but they never co-exist in any sessions. Such an incompatible relation unveils the fact that Stylus 2nd Generation is not supported by Tablet 1st Generation (i.e., version incompatibility) without a knowledge graph. Similarly, Stylus 1st Generation and Stylus 2nd Generation are also of an incompatible relation. With an incompatible relation, observing one item in the current session suggests to lower the probability of recommending the other. We provide a case study in Section 4.6 to help understand how these relations affect the odds of recommending different items.

We summarize our main technical contributions as follows.

- To the best of our knowledge, this is the first paper that proposes to learn multi-faceted global item relations to ameliorate session-based recommendation. The relations are learned in a session-independent and data-driven manner, making it highly efficient and applicable to all datasets without requiring any additional prior knowledge (e.g., knowledge graphs).
- We identify three types of global item relations that are advantageous to session-based recommendation, and propose different graph construction methods to effectively extract these relations. In particular, we show that negative relations (e.g., incompatible relations) that have not been studied before are critical to model performance.
- We propose a novel solution based on multi-faceted global item relation learning (MGIR) to encode different relations

using different aggregation layers and generate enhanced session representations by fusing positive and negative relations. Our solution is flexible to accommodate new item relations and can easily integrate different existing methods to generate better representations from global relation enhanced session information.
- We conduct extensive experiments on three public benchmark datasets and show that our MGIR model consistently outperforms a large number of state-of-the-art competitors.

2 RELATED WORK

Since session-based recommendation can be seen as a special case of sequential recommendation, in this section we review both sequential recommendation models and session-based recommendation models.

2.1 Sequential Recommendation

The key idea of sequential recommendation is to explicitly exploit the temporal order of users' historical interactions for next-item prediction. Early research makes use of Markov chains (MC) to mine sequential patterns in historical data. FPMC [27] models users' long-term preferences by combining first-order MC and matrix factorization to capture both sequential patterns and long-term user preference. Caser [32] considers CNNs as the backbone network to embed recent items into an "image" so as to learn sequential patterns as local features of the image using convolutional filters. RCNN [46] combines CNNs and long-short term memory networks (LSTMs) to capture long-term dependencies and short-term sequential patterns. HGN [21] models long-term and short-term interests via feature gating and instance gating modules. DMAN [30] segments long behavior sequences into a series of sub-sequences and maintains a set of memory blocks to preserve long-term interests.

Attention mechanisms [34] are another choice for sequential recommendation. SASRec [15] captures sequential dynamics based on self attention. MARank [49] takes into consideration union-level item interactions and uses a multi-order attentive model to fuse union-level and individual-level information. Liu *et al.* [18] propose a non-invasive self-attention mechanism to incorporate side information into sequential recommendation. RetaGNN [12] proposes a relational temporal attentive graph neural network to simultaneously accommodate conventional, inductive and transferable settings. There are also studies [36, 37, 40] that leverage knowledge graphs to improve sequential recommendation. Some very recent studies further consider useful temporal information beyond temporal order (e.g., time intervals between historical interactions and time intervals between recommendation timestamps and historical interaction timestamps) to model more complicated user behavior [4, 6, 10, 48].

2.2 Session-Based Recommendation

Unlike sequential recommendation, user profiles and long-term interactions are not available to session-based recommendation. Therefore, how to more effectively represent a session lies in the core of session-based recommendation. GRU4Rec [11] is the first model that adopts gated recurrent units (GRUs) to model the sequentiality of item interactions. NARM [17] extends GRU4Rec by introducing a hybrid encoder with an attention mechanism to extract the main intent in the current session while modeling its sequential behavior. STAMP [19] is equipped with a short-term attention/memory priority model to capture a user's general and current interests.

Recently, GNN-based methods have become the mainstream. SR-GNN [42] is a seminal GNN-based method that transforms a session into a directed unweighted graph and that utilizes gated GNNs to generate the session representation. GC-SAN [45] improves SR-GNN with the self-attention mechanism to capture long-range dependencies among items. Qiu *et al.* [26] creatively formulate session-based recommendation as a graph classification problem, and design a weighted attention graph layer and a readout function to learn embeddings of items and sessions. Chen and Wong [5] identify two information loss problems of GNN-based methods, including the lossy session encoding problem and the ineffective long-range dependency capturing problem. SGNN-HN [24] introduces a star node to each session graph in order to capture long-distance relations among items. Meng *et al.* [22] propose to enhance session representations by utilizing users' micro-behavior and item attributes as side information. Cho *et al.* [7] propose to imitate users' general interests by modeling proxies of sessions. While these studies have obtained encouraging results, learning from short sessions independently may be inherently inadequate to accurately unveil users' real intents.

A new line of research has started to explore collaborative information from other sessions to enhance the current session's representation. RNN-KNN [14] selects k nearest neighbor sessions by calculating the cosine similarity scores between the sessions' embeddings. STAN [9] further enhances the selection of k nearest neighbor sessions by incorporating sequential and temporal information. CSRM [39] proposes to enhance a session's representation by considering the latest m sessions. CoSAN [20] combines neighbor sessions according to the Jaccard similarity coefficients between the current session and other sessions. However, as mentioned before, this line of research generally suffers from the ad-hoc notions of relatedness. More recent studies are based on a global graph built from all sessions in training data. GCE-GNN [41] proposes to enhance session-level representations by using session-aware item co-occurrence relations. Qiu *et al.* [25] introduce the notion of broadly connected session (BCS) graphs, which expands a session graph by including n-hop neighbors in the global graph. CA-TCN [47] builds both a global item graph and a session-context graph to model cross-session influence on both items and sessions. MTD [13] utilizes graphical mutual information maximization to capture global item-wise transition information to enhance the current session's representation. S^2-DHCN [44] utilizes hypergraph convolutional networks to capture high-order item relations within individual sessions and uses self-supervised learning to enhance session representations. With a different motivation, DAT-MDI [3] combines dual transfer with graph neural networks to learn cross-domain representations for session-based recommendation. All the above studies, however, neglect the fact that there exist multiple types of item relations that can together substantially better session representations. Hence, we propose a new research direction to learn multi-faceted session-independent global item relations, including negative relations that have not been studied before.

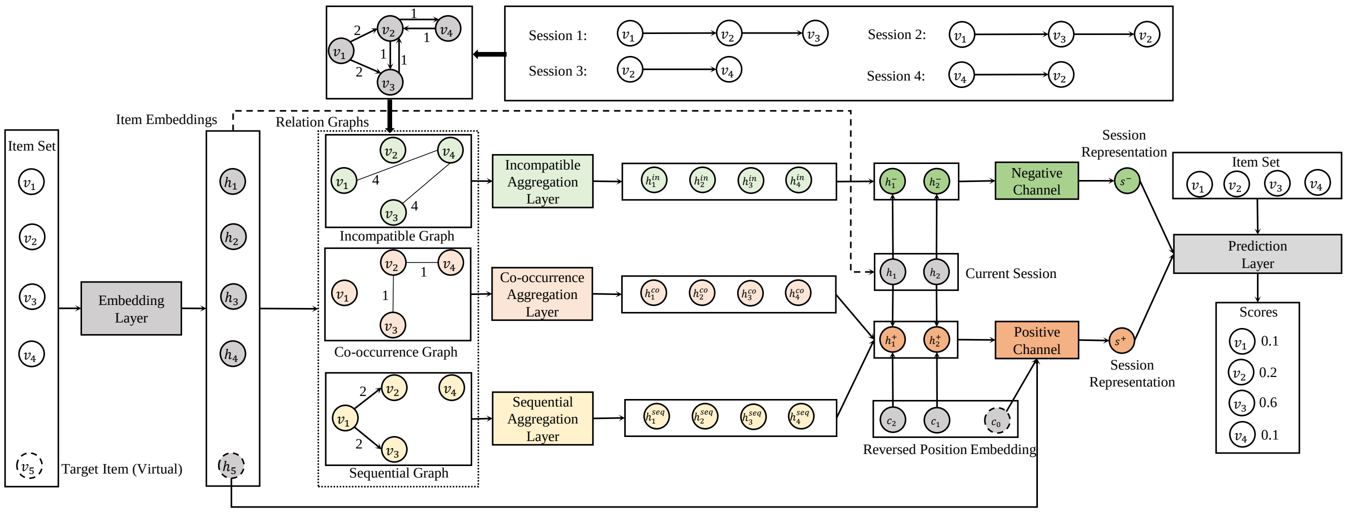
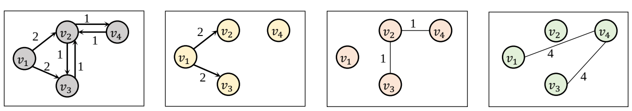


Figure 2: The overall architecture of the proposed MGIR model.



(a) Base (b) Sequential (c) Co-occurrence (d) Incompatible

Figure 3: An illustration of the construction of multi-faceted global item relation graphs.

3 METHODOLOGY

Our MGIR model takes as input a set of *anonymous* sessions $\mathcal{S} = \{S_1, S_2, \cdots, S_{|\mathcal{S}|}\}$ over the item universe $\mathcal{I} = \{i_1, i_2, \cdots, i_{|\mathcal{I}|}\}$ and the current session $S = \{s_1, s_2, \cdots, s_l\}$, where $s_i \in \mathcal{I}$ is the ith item interacted in the current session, and outputs the top-K items from $\mathcal{I}$ that are most likely to be interacted in the next time step.

As illustrated in Figure 2, MGIR consists of four major components, including the construction of multi-faceted global item relation graphs, the graph aggregation layers to generate global item embeddings for each type of relations, the session representation learning module to generate positive and negative session representations and the final prediction layer. In the following, we detail each component.

3.1 Global Item Relation Graph Construction

3.1.1 Global Base Graph. All global relation graphs are derived from a global base graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ constructed from all sessions in training data, which is a weighted directed graph with $\mathcal{V} = \mathcal{I}$ being the set of all items and $\mathcal{E} = \{\varepsilon_{ij}\}$ the set of edges. There is an edge ε_{ij} from v_i to v_j if v_j is observed after v_i in some sessions. It is worth noting that we also capture the transition information between non-adjacent items in order to fully capture item transition patterns. Each edge ε_{ij} is associated with a weight that is the frequency of v_i being followed by v_j in all sessions. Given the sessions in Figure 1, Figure 3(a) shows the corresponding global base graph.

3.1.2 Sequential Relation Graph. Sequential relations are probably the most important item relations for session-based recommendation. Unlike existing methods that normally assume overly strong sequential relations between adjacent items in individual sessions, we propose a purely data-driven approach to identify *only* reliable sequential relations observed in *all* sessions. Specifically, we consider two items to have a sequential relation only if they appear in a particular order in a sufficient number of sessions. Mathematically, there is an edge ε_{ij}^{seq} from v_i to v_j in the sequential relation graph $\mathcal{G}^{seq}$ iff

$$w_{ij} > (1+\eta)w_{ji} \text{ and } w_{ij} + w_{ji} > \mu,$$

where w_{ij} is the weight of ε_{ij} and w_{ji} is the weight of ε_{ji} in the global base graph $\mathcal{G}$, $\eta \geq 0$ is a ratio hyper-parameter used to distinguish between sequential (i.e., $w_{ij} \gg w_{ji}$) and co-occurrence relations (i.e., $w_{ij} \approx w_{ji}$), and $\mu \geq 0$ is a threshold hyper-parameter used to filter out unreliable relations. The introduction of μ is inspired by the minimum support threshold used in frequent itemset mining [1, 2], and it represents an interesting trade-off between the quality and quantity of captured relations: a smaller μ could introduce more sequential relations with more noise (e.g., unreliable relations derived from long-tail items). We study this trade-off in our experiments.

The weight w_{ij}^{seq} of ε_{ij}^{seq} is $w_{ij} - w_{ji}$, indicating the sequential frequency between the two items. The existence of ε_{ij}^{seq} suggests that v_i is more likely to be followed by v_j in a session and that implicitly v_j is less likely to be followed by v_i in a session. Similarly, $\mathcal{G}^{seq}$ is a weighted directed graph. Figure 3(b) shows the construction of the sequential relation graph $\mathcal{G}^{seq}$ based on the global base graph in Figure 3(a).

3.1.3 Co-Occurrence Relation Graph. Different from $\mathcal{G}^{seq}$, the co-occurrence graph $\mathcal{G}^{co}$ simply indicates that two items are more likely to co-exist in sessions, but not in any particular order. Therefore, $\mathcal{G}^{co}$ is a weighted *undirected* graph. Similarly, we use a data-driven way to derive co-occurrence relations. There is an edge ε_{ij}^{co} between v_i and v_j in the co-occurrence graph $\mathcal{G}^{co}$ iff

$$\max(w_{ij}, w_{ji}) \le (1+\eta)\min(w_{ij}, w_{ji}) \text{ and } w_{ij} + w_{ji} > \mu.$$

Here the hyper-parameters η and μ play similar roles as in sequential relations. The weight w_{ij}^{co} of ε_{ij}^{co} is defined as $\min(w_{ij}, w_{ji})$, which indicates the co-occurrence frequency of the two items. Figure 3(c) gives the co-occurrence relation graph $\mathcal{G}^{co}$ resulted from the global base graph in Figure 3(a).

3.1.4 Incompatible Relation Graph. Unlike co-occurrence and sequential relations that can be explicitly observed in data, learning dependable incompatible relations renders non-trivial technical challenges and needs a careful design. For example, one could define an incompatible relation between two items because they never co-exist in any sessions, which well illustrates two common challenges of defining incompatible relations. First, due to the prevalent data sparsity issue, long-tail items with few interactions would form many erroneous incompatible relations. Second, it is not possible to capture the strength of such relations, i.e., to what extent the existence of an item lowers the odds of having another item in the same session. In addressing these issues, our general design principle is to resort to more strict *observable* conditions to capture *only* reliable incompatible relations.

To this end, we first rank all items according to their numbers of occurrences in all sessions and define the last δ of items as the long-tail items, where $\delta \in [0, 1]$ is another threshold hyper-parameter. We denote the set of long-tail items by $\overline{\mathcal{V}}$. To avoid noise introduced by long-tail items, we define incompatible relations on only non-long-tail items. Then we establish incompatible relations between items by considering whether they ever share same contexts. Intuitively, two items are incompatible if they frequently co-exist with other items (e.g., contexts), but never co-exist with each other. To capture this intuition, we convert the global base graph $\mathcal{G}$ to an undirected graph $\widetilde{\mathcal{G}}$ with $\widetilde{\mathcal{V}} = \mathcal{V}$, $\widetilde{\mathcal{E}} = \mathcal{E}$, and $\tilde{w}_{ij} = w_{ij} + w_{ji}$, where $\tilde{w}_{ij}$ is the weight between v_i and v_j in the new undirected graph $\widetilde{\mathcal{G}}$. For items v_i and v_j, we define their context set as

$$\mathcal{B}_{ij} = \{b | \tilde{w}_{ib} > \mu, \tilde{w}_{jb} > \mu, b \in \widetilde{\mathcal{V}}\},$$

where each b is a common neighbor of v_i and v_j in $\widetilde{\mathcal{G}}$, and μ is the same threshold hyper-parameter used in the construction of the sequential relation graph and co-occurrence relation graph to filter out unreliable relations.

There is an edge ε_{ij}^{in} between v_i and v_j in the incompatible relation graph $\mathcal{G}^{in}$ iff

$$v_i \notin \overline{\mathcal{V}} \text{ and } v_j \notin \overline{\mathcal{V}} \text{ and } \mathcal{B}_{ij} \neq \emptyset \text{ and } \tilde{w}_{ij} = 0.$$

Here $\tilde{w}_{ij} = 0$ indicates that v_i and v_j never co-exist in any sessions. Finally, we define the weight w_{ij}^{in} of ε_{ij}^{in} in the incompatible relation graph $\mathcal{G}^{in}$ as $w_{ij}^{in} = \sum_{b \in \mathcal{B}_{ij}} (\tilde{w}_{ib} + \tilde{w}_{jb})$. In general, the more contexts two items share, the more reliable and stronger their incompatible relation is. We empirically study the impact of η, μ, and δ in Section 4.4. Figure 3(d) illustrates an incompatible relation graph $\mathcal{G}^{in}$. It is worth noting that the three types of global item relations are exclusive in that any two items can have *at most* one type of relations.

3.2 Graph Aggregation Layers

With the constructed graphs, we design different graph aggregation layers to generate items' global embeddings under each relation graph. We generate an item v_i's base embedding by mapping its ID into a dense embedding vector $\mathbf{h}_i \in \mathbb{R}^d$, where d is the dimension of the embedding vector. More specifically, we build a parameter matrix as an embedding look-up table for embedding initialization:

$$\mathbf{h}_i = \mathbf{x}_i \mathbf{W}_h, \tag{1}$$

where $\mathbf{W}_h$ is a trainable matrix, and $\mathbf{x}_i$ is the one-hot encoding of v_i's ID.

3.2.1 Sequential Graph Aggregation Layer. Let $\mathcal{P}_i^{seq}$ be the set of incoming neighbors of v_i in the sequential relation graph $\mathcal{G}^{seq}$ and $\mathcal{Q}_i^{seq}$ the set of outgoing neighbors. The neighbor information of v_i is aggregated over $\mathcal{P}_i^{seq}$ and $\mathcal{Q}_i^{seq}$ via

$$\begin{aligned} \mathbf{p}_i^{seq} &= \sum_{v_j \in \mathcal{P}_i^{seq}} w_{ji}^{seq} \mathbf{h}_j, \\ \mathbf{q}_i^{seq} &= \sum_{v_j \in \mathcal{Q}_i^{seq}} w_{ij}^{seq} \mathbf{h}_j, \end{aligned} \tag{2}$$

where w_{ij}^{seq} and w_{ji}^{seq} are the normalized weights of edges ε_{ij}^{seq} and ε_{ji}^{seq} in $\mathcal{G}^{seq}$.

Inspired by the graph attention network (GAT) [35], we utilize an attention mechanism and a bilinear function to calculate attention weights for v_i's neighbor sets (i.e., incoming and outgoing neighbors) as

$$\begin{aligned} \alpha_{\mathcal{P}_i}^{seq} &= \frac{\sigma\left(\mathbf{p}_i^{seq} \mathbf{W}_{\mathcal{P}}^{seq} \mathbf{h}_i^{\top}\right)}{\sqrt{d}}, \\ \alpha_{\mathcal{Q}_i}^{seq} &= \frac{\sigma\left(\mathbf{q}_i^{seq} \mathbf{W}_{\mathcal{Q}}^{seq} \mathbf{h}_i^{\top}\right)}{\sqrt{d}}, \end{aligned} \tag{3}$$

where $\sigma(\cdot)$ is the ReLU function, and $\mathbf{W}_{\mathcal{P}}^{seq}, \mathbf{W}_{\mathcal{Q}}^{seq} \in \mathbb{R}^{d \times d}$ are trainable parameter matrices.

Then, we adaptively combine the neighbor information of incoming and outgoing neighbors via

$$\mathbf{x}_i^{seq} = \alpha_i^{seq} \mathbf{p}_i^{seq} + (1 - \alpha_i^{seq}) \mathbf{q}_i^{seq}, \tag{4}$$

$$\alpha_i^{seq} = \frac{\exp(\alpha_{\mathcal{P}_i}^{seq})}{\exp(\alpha_{\mathcal{P}_i}^{seq}) + \exp(\alpha_{\mathcal{Q}_i}^{seq})}. \tag{5}$$

Note that $\mathbf{x}_i^{seq}$ only contains information from v_i's neighbors. Thus, we further aggregate the base embedding $\mathbf{h}_i$ to generate the global sequential embedding of v_i. Specifically, instead of using common aggregation methods, such as mean pooling, sum pooling and multi-layer perceptron (MLP), we resort to a convolution operation, which is inspired by ConvKB [23]. In our context, a convolution operation well preserves dimension-wise information without introducing additional parameters. Empirically, it achieves consistent performance improvements on different datasets. Formally, the global sequential embedding of v_i is

$$\mathbf{h}_i^{seq} = Conv^{seq}(\mathbf{h}_i \parallel \mathbf{x}_i^{seq}), \tag{6}$$

where $Conv^{seq}(\cdot)$ is a convolution operation with filter size 2×1 and stride 1, and $\|$ is the concatenation operation.

3.2.2 Co-Occurrence Graph Aggregation Layer. Different from $\mathcal{G}^{seq}$, the co-occurrence relation graph $\mathcal{G}^{co}$ is an *undirected* graph. Without the need to distinguish incoming and outgoing neighbors, we denote the set of v_i's neighbors in $\mathcal{G}^{co}$ by $\mathcal{N}_i^{co}$. Therefore, the information aggregation process on $\mathcal{N}_i^{co}$ is simply done via

$$\mathbf{x}_i^{co} = \sum_{v_j \in \mathcal{N}_i^{co}} w_{ij}^{co} \mathbf{h}_j, \tag{7}$$

where w_{ij}^{co} is the normalized weight of edge ε_{ji}^{co} in $\mathcal{G}^{co}$. Similar to Eq. (6), we use a convolution operation to fuse $\mathbf{x}_i^{co}$ with the base embedding $\mathbf{h}_i$ as follows:

$$\mathbf{h}_i^{co} = Conv^{co}(\mathbf{h}_i \parallel \mathbf{x}_i^{co}), \tag{8}$$

where $Conv^{co}(\cdot)$ is also a convolution operation with filter size 2×1 and stride 1.

3.2.3 Incompatible Graph Aggregation Layer. Since the incompatible relation graph is also an undirected graph, we leverage a similar information aggregation process over v_i's neighbors $\mathcal{N}_i^{in}$ in the incompatible graph $\mathcal{G}^{in}$ as follows:

$$\mathbf{x}_i^{in} = \sum_{v_j \in \mathcal{N}_i^{in}} w_{ij}^{in} \mathbf{h}_j, \tag{9}$$

where w_{ji}^{in} is the normalized weight of edge ε_{ji}^{in} in $\mathcal{N}_i^{in}$.

Compared with the sequential and co-occurrence relation graphs, the incompatible relation graph $\mathcal{G}^{in}$ has more isolated nodes. To better prevent the noise due to these isolated nodes, we use a mean pooling operation to generate the final incompatible item embedding as

$$\mathbf{h}_i^{in} = Mean(\mathbf{x}_i^{in} \parallel \mathbf{h}_i). \tag{10}$$

3.3 Session Representation Learning

With the global embeddings generated by the above aggregation layers, we are ready to learn the representations of sessions from both global-level and session-level information. Recall that global sequential and co-occurrence embeddings suggest which items are more likely to be the next item while incompatible embeddings suggest which items are less likely to be the next item. Consequently, instead of generating a single representation for a session, we generate both *positive* and *negative* session representations. Given an item v_i, its positive and negative global embeddings are generated via

$$\mathbf{h}_i^{+} = \mathbf{h}_i^{seq} + \mathbf{h}_i^{co}, \mathbf{h}_i^{-} = \mathbf{h}_i^{in}. \tag{11}$$

To further leverage the information captured in the base embeddings, we design a gated mechanism to fuse $\mathbf{h}_i$ with $\mathbf{h}_i^+$ and $\mathbf{h}_i^-$. The first step is to learn the weights for $\mathbf{h}_i^+$ and $\mathbf{h}_i^-$ as follows:

$$\begin{aligned} \alpha_i^{+} &= \sigma(\mathbf{h}_i^{+}\mathbf{W}^{+} + \mathbf{h}_i\mathbf{W}), \\ \alpha_i^{-} &= \sigma(\mathbf{h}_i^{-}\mathbf{W}^{-} + \mathbf{h}_i\mathbf{W}), \end{aligned} \tag{12}$$

where $\sigma(\cdot)$ is the sigmoid function and $\mathbf{W}^+, \mathbf{W}^-, \mathbf{W} \in \mathbb{R}^{d \times d}$ are trainable parameter matrices. Then we implement the gated mechanism as

$$\begin{aligned} \mathbf{h}_i^{+} &= \alpha_i^{+}\mathbf{h}_i^{+} + (1 - \alpha_i^{+})\mathbf{h}_i, \\ \mathbf{h}_i^{-} &= \alpha_i^{-}\mathbf{h}_i^{-} + (1 - \alpha_i^{-})\mathbf{h}_i. \end{aligned} \tag{13}$$

3.3.1 Positive Session Representation Learning Layer. Modeling the short-term intent of the current session is key to its representation. Previous works [20, 42, 45] simply consider the last interacted item as the session's current interest. However, this assumption may not always hold in practice [51]. In this paper, we introduce a virtual target item to better depict the session's short-term intent. We assign a special ID $|\mathcal{I}| + 1$ to the target item. Then its embedding $\tilde{\mathbf{h}} \in \mathbb{R}^d$ can be learned in a similar way to other items.

To factor the sequentiality in a session, we make use of reversed position embeddings $\mathbf{c}_i \in \mathbb{R}^d$ suggested by GCE-GNN [41]. Since sessions are of different lengths, using reversed position embeddings can more accurately capture the influence of each item on the next item to recommend. Reversed position embeddings can be similarly learned from one-hot vectors and a trainable parameter matrix. We denote the reversed position embedding of an item v_i in the current session S by $\mathbf{c}_{|S|-i}$. The reversed position embedding of the target item is then $\mathbf{c}_0$. Next, we enhance each item in the session S, including the target item, by integrating its reversed position embedding as follows:

$$\begin{aligned} \tilde{\mathbf{h}} &= \tanh\left([\tilde{\mathbf{h}} \| \mathbf{c}_0]\mathbf{W}_c\right), \\ \mathbf{h}_i^{+} &= \tanh\left([\mathbf{h}_i^{+} \| \mathbf{c}_{|S|-i}]\mathbf{W}_c\right), \end{aligned} \tag{14}$$

where $\mathbf{W}_c \in \mathbb{R}^{2d \times d}$ is a trainable parameter matrix shared by all items.

Since the target item well represents the learned intent of the current session, we employ a soft-attention mechanism to measure the relevance between the target item and other items via

$$\beta_i^{+} = \sigma(\mathbf{h}_i^{+}\mathbf{W}_1^{+} + \tilde{\mathbf{h}}\mathbf{W}_2^{+})\mathbf{W}_3^{+}, \tag{15}$$

where $\sigma(\cdot)$ is the sigmoid function, $\mathbf{W}_1^+, \mathbf{W}_2^+ \in \mathbb{R}^{d \times d}$ and $\mathbf{W}_3^+ \in \mathbb{R}^{d \times 1}$ are trainable parameter matrices. Finally, we generate the positive session representation via

$$\mathbf{s}^{+} = \sum_{v_i \in S} \beta_i^{+}\mathbf{h}_i^{+}. \tag{16}$$

3.3.2 Negative Session Representation Learning Layer. Different from positive session representations, negative session representations mainly rely on the global incompatible item relations, which are undirected. Therefore, we do not consider reversed position embeddings and the target item in the negative session representation learning layer. We apply an attention mechanism to calculate each item's contribution to the negative session representation via

$$\beta_i^{-} = \sigma(\mathbf{h}_i^{-}\mathbf{W}_1^{-} + \mathbf{h}_{|S|}^{-}\mathbf{W}_2^{-})\mathbf{W}_3^{-}, \tag{17}$$

where $\mathbf{h}_{|S|}^-$ is the last item of the current session, and $\mathbf{W}_1^-, \mathbf{W}_2^- \in \mathbb{R}^{d \times d}$ and $\mathbf{W}_3^- \in \mathbb{R}^{d \times 1}$ are trainable parameter matrices. After that, we can generate the negative session representation via

$$\mathbf{s}^{-} = \sum_{v_i \in S} \beta_i^{-}\mathbf{h}_i^{-}. \tag{18}$$

3.4 Prediction and Model Optimization

After obtaining both positive and negative session representations, we make the next-item recommendation by computing a probability distribution of the next item over the entire item universe. For each

Table 1: Properties of the datasets used in our experiments.

Dataset	# of train	# of test	# of items	Avg. length
Tmall	351,268	25,898	40,727	6.69
RetailRocket	433,643	15,132	36,968	5.43
Last.fm	2,837,330	672,833	38,615	11.78

candidate item $v_i \in \mathcal{V}$, we can calculate its relevance to the current session via

$$z_i = \mathbf{s}^{+}\mathbf{h}_i^{\top} - \sigma(\mathbf{s}^{-}\mathbf{h}_i^{\top}), \tag{19}$$

where $\sigma(\cdot)$ is the LeakyReLU function with hyper-parameter γ. Here LeakyReLU is used to better balance the effect from negative scores. Then the predicted probability of the next item being v_i, $\hat{y}_i$, can be computed by:

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_{v_j \in \mathcal{V}} \exp(z_j)}. \tag{20}$$

Finally, we train our model by minimizing the cross-entropy of the prediction results $\hat{y}_i$:

$$\mathcal{L}(\mathrm{y}, \tilde{\mathrm{y}}) = -\sum_{i=1}^{|\mathcal{V}|} (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)) + \lambda \|\Theta\|_2^2, \tag{21}$$

where $\mathbf{y}$ denotes the one-hot encoding vector of the ground truth item, λ is a hyper-parameter controlling the strength of L_2 regularization, and Θ is the set of model parameters.

4 EXPERIMENTS

In this section, we perform comprehensive experiments on the proposed MGIR model and a large number of state-of-the-art models over three public real-word datasets. Our experiments are devised to answer the following key research questions:

- **RQ1**: Does MGIR outperform other state-of-the-art session-based recommendation methods?
- **RQ2**: How do different global item relation graphs in MGIR contribute to the performance?
- **RQ3**: How do different hyper-parameter settings (i.e., η, μ, and δ) affect MGIR's performance?
- **RQ4**: How does MGIR perform under different methods of fusing base embeddings and global item embeddings?

In addition, we provide a case study to show how multi-faceted global item relations affect the odds of recommending different items.

4.1 Experimental Settings

4.1.1 Datasets. To evaluate the effectiveness of MGIR, we conduct experiments on three public benchmark datasets widely used in the literature: *Tmall*[1], *RetailRocket*[2] and *Last.fm*[3], which are of different properties. *Tmall* is a dataset from the IJCAI-15 competition. It contains a large number of anonymous shopping logs on the Tmall online shopping platform. *RetailRocket* is a dataset from a Kaggle contest, which includes users' six-month browsing histories on an e-commerce website. *Last.fm* has users' music listening histories, in which artists are considered as items. The properties of the three datasets are summarized in Table 1.

Following previous works [26, 41, 42, 44], we filter out short sessions with less than 2 items and infrequent items (i.e., items whose frequency is less than 5) and apply the data augmentation technique described in [31, 41, 42]. For example, for an input session $S = [s_1, s_2, \cdots, s_l]$, we generate a series of sequences and corresponding labels as the augmented input by a sequence splitting preprocessing step, i.e., $([s_1], s_2), ([s_1, s_2], s_3), \cdots, ([s_1, s_2, \cdots, s_{l-1}], s_l)$. We also divide training and test data by time to avoid potential data leakage: the most recent data (i.e., last week or most recent 20% sessions) are used as the test data, and the remaining historical data as the training data. We consider Precision@20 (P@20 for short) and MRR@20 as the evaluation metrics, both of which are widely used in [13, 19, 41, 42].

4.1.2 Competing Models. We compare MGIR with a wide range of representative methods:

- **FPMC** [27] is a Markov-chain based method for sequential recommendation. We omit the user information since it is not available in session-based recommendation.
- **GRU4Rec**[4] [11] is an RNN-based method that uses gated recurrent units (GRUs) to capture sequential information and model the short-term intent underlying the current session.
- **NARM**[5] [17] uses a hybrid encoder with an attention mechanism to extract the main intent in the current session and model the user's sequential behavior.
- **STAMP**[6] [19] is equipped with a short-term attention/memory priority model to capture a user's general interest and current interest.
- **SR-GNN**[7] [42] is a seminal GNN-based method that transforms a session into a directed unweighted graph and that utilizes gated GNNs to generate the session representation.
- **FGNN**[8] [26] converts the current session into a directed weighted graph and uses an adapted graph attention network [35] to learn item representations.
- **GCE-GNN**[9] [41] is a state-of-the-art GNN-based model that constructs a global co-occurrence graph from all sessions to learn global information of items and that integrates global information by considering its similarity to a rough session representation.
- **MTD**[10] [13] constructs a global graph which connects adjacent items in each session, and utilizes graphical mutual information maximization to capture global item-wise transition information in order to enhance the current session's representation.
- S^2**-DHCN**[11] [44] utilizes hypergraph convolutional networks to capture high-order item relations beyond pairwise ones

[1]https://tianchi.aliyun.com/dataset/dataDetail?dataId=42
[2]https://www.kaggle.com/retailrocket/ecommerce-dataset
[3]http://ocelma.net/MusicRecommendationDataset/lastfm-1K.html

[4]https://github.com/hidasib/GRU4Rec
[5]https://github.com/lijingsdu/sessionRec_NARM
[6]https://github.com/uestcnlp/STAMP
[7]https://github.com/CRIPAC-DIG/SR-GNN
[8]https://github.com/RuihongQiu/FGNN
[9]https://github.com/CCIIPLab/GCE-GNN
[10]https://github.com/sessionRec/MTD
[11]https://github.com/xiaxin1998/DHCN

Table 2: Experimental results (%) on the three datasets. The best results are boldfaced and the second-best results are underlined. All improvements are significant with p-value < 0.05 based on paired t-tests.

	Tmall		RetailRocket		Last.fm	
	Precision@20	MRR@20	Precision@20	MRR@20	Precision@20	MRR@20
FPMC (*WWW'10*)	16.06	7.32	32.37	13.82	12.86	3.78
GRU4Rec (*ICLR'16*)	10.93	5.89	44.01	23.67	17.90	5.39
NARM (*CIKM'17*)	23.30	10.70	50.22	24.59	21.83	7.59
STAMP (*KDD'18*)	26.47	13.36	50.96	25.17	22.01	7.98
SR-GNN (*AAAI'19*)	27.57	13.72	50.32	26.57	22.33	8.23
FGNN (*CIKM'19*)	25.24	10.39	50.20	25.89	22.20	8.02
GCE-GNN (*SIGIR'20*)	<u>33.42</u>	<u>15.42</u>	<u>54.58</u>	<u>28.09</u>	<u>24.39</u>	<u>8.63</u>
MTD (*AAAI'21*)	29.12	13.73	53.46	26.36	22.60	7.78
S^2-DHCN (*AAAI'21*)	31.42	15.05	53.66	27.30	22.06	7.57
MGIR	**36.41**	**17.42**	**56.62**	**29.84**	**24.72**	**8.82**
Improvement	8.95%	12.97%	3.74%	6.23%	1.35%	2.20%
p-value	$9e^{-4}$	$1e^{-4}$	$8e^{-4}$	$1e^{-3}$	$3e^{-4}$	$1e^{-3}$

within individual sessions and uses self-supervised learning to enhance session representations.

To make a fair comparison, we focus on the models whose source code is publicly available. We omit CSRM [39] and CoSAN [20] because GCE-GNN and MTD have already reported better performance.

4.1.3 Implementation Details. Identical to the settings of previous methods [17, 19, 41, 42, 44], the embedding size is fixed to 100 for all methods, and the embedding parameters are initialized with a Gaussian distribution. We optimize MGIR with Adam [16] and use the default learning rate of 0.001 and default mini-batch size of 100 (we increase the mini-batch size on Last.fm to 512 to improve training speed). A learning rate decay strategy is adopted, which decreases the learning rate by 90% after every 3 epochs. The dropout rate is searched in the range of $\{0.1, 0.2, \cdots, 0.5\}$ to avoid overfitting in graph aggregation layers and session representation learning layers. The L_2 regularization coefficient λ is set to 10^{-5} on Tmall and 10^{-6} on the other two datasets. We tune all hyper-parameters on a validation dataset which is a random 10% subset of the training dataset. The hyper-parameter γ of $LeakyReLU(\cdot)$ is searched in the range of $\{0, 0.1, 0.2, \cdots, 1\}$. The ratio hyper-parameter η, threshold hyper-parameter μ and threshold hyper-parameter δ are searched in the ranges of $\{0, 0.5, 1.0, \cdots, 3.5\}$, $\{0, 10, 30, 50, 70, 90, 100\}$ and $\{0.8, 0.825, 0.85, \cdots, 0.975\}$, respectively. In particular, the search range of δ, which is used to identify long-tail items, is determined by the 20/80 principle of the long-tail distribution [29]. The hyper-parameters of all competing models are carefully tuned by grid search, and the best performances are reported.

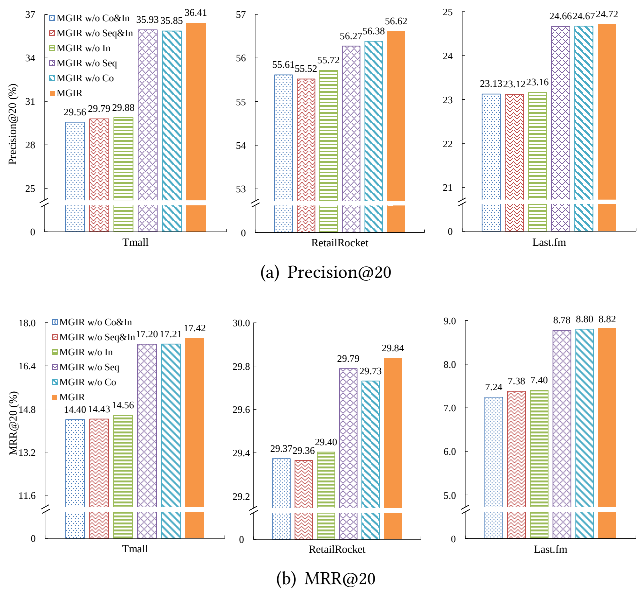
We implement our model in PyTorch 1.7.0 and Python 3.8.3. All experiments were run on a workstation with an Intel Xeon Platinum 2.40GHz CPU, a NVIDIA Quadro RTX 8000 GPU and 500GB RAM. The code and preprocessed data are publically available at https://github.com/zc-97/MGIR.

4.2 Overall Performance Comparison (RQ1)

We report the main experimental results in Table 2, where the best results are boldfaced and the second-best results are underlined. We can draw a few interesting observations.

- MGIR consistently yields the best performance on all three datasets. In particular, its relative improvements over the strongest baselines are 8.95%, 3.74% and 1.35% in terms of Precision@20 and 12.97%, 6.23% and 2.20% in terms of MRR@20 on Tmall, RetailRocket and Last.fm, respectively. All improvements are significant with p-value < 0.05. The statistical significance tests were conducted by performing paired t-tests. Such results generally demonstrate the superiority of our solution.
- Compared with the methods based on individual sessions, the methods considering collaborative information from other sessions consistently improve the performance by a significant margin, which confirms the inherent inadequacy of learning from only individual sessions.
- Among the methods considering collaborative information from other sessions, MGIR consistently achieves the best performance. We attribute such improvements to learning multi-faceted session-independent global item relations, which is able to better distinguish the true relations between different pairs of items and thus generate more representative session embeddings. In particular, negative relations (e.g., incompatible relations) that have not been studied before are beneficial as we will show later.

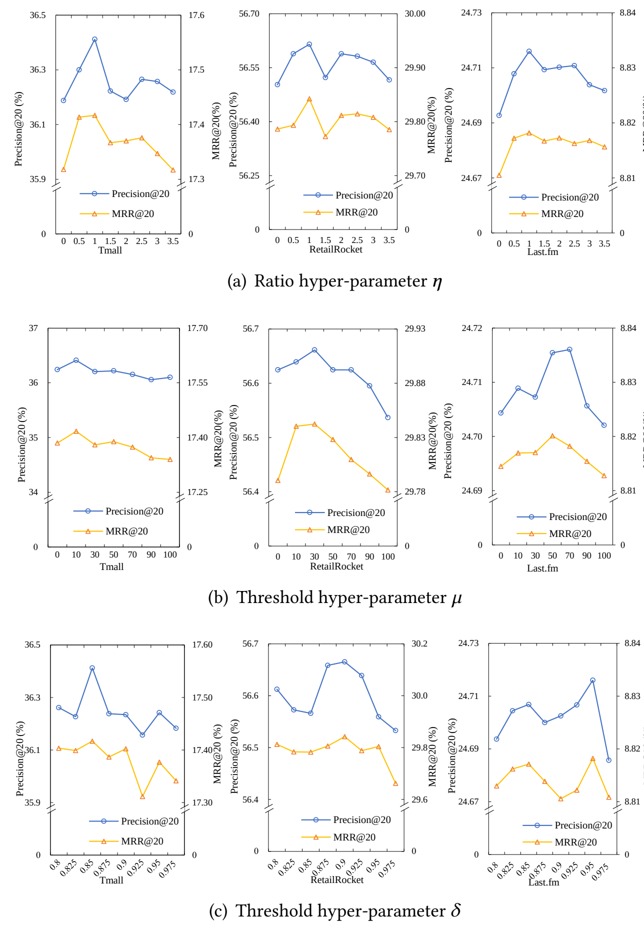
Running Time. To show the practical applicability of MGIR, we also report the running time. Here we focus on the strongest methods based on global graphs. The running time of an epoch in training of MTD, MGIR, GCE-GNN and S^2-DHCN is about 70s, 278s, 175s and 1,063s for Tmall, 456s, 840s, 1,482s and 1,749s for RetailRocket, and 351s, 508s, 808s and 9,314s for Last.fm, respectively. Such results show that, due to learning global item relations in a



(a) Precision@20

(b) MRR@20

Figure 4: Impact of different global item relation graphs.



(a) Ratio hyper-parameter η

(b) Threshold hyper-parameter μ

(c) Threshold hyper-parameter δ

Figure 5: Impact of different hyper-parameters.

session-independent way, MGIR can achieve better performance with lower or comparable complexity.

4.3 Impact of Global Relation Graphs (RQ2)

To verify the contributions of different global item relation graphs in MGIR, we conduct an ablation study with several variants:

- MGIR w/o Co&In: MGIR with only the sequential graph.
- MGIR w/o Seq&In: MGIR with only the co-occurrence graph.
- MGIR w/o Seq: MGIR without the sequential graph.
- MGIR w/o Co: MGIR without the co-occurrence graph.
- MGIR w/o In: MGIR without the incompatible graph.

Figure 4 shows the performances of different variants in term of Precision@20 and MRR@20. It can be observed that each relation graph contributes to performance. The incompatible relation graph is particularly important to achieve encouraging results, justifying one of our main technical contributions. Furthermore, combining multiple relation graphs can improve performance in all cases, which confirms the rationality of learning multi-faceted global item relations for session-based recommendation.

4.4 Impact of Hyper-Parameters (RQ3)

We further study the impact of the key hyper-parameters η, μ and δ, and report the results in Figure 5. On all datasets, we can find that: (1) the best results are achieved when $\eta = 1$. This is because a larger η value will miss more useful sequential relations while a smaller η value will not be able to well distinguish sequential and co-occurrence relations. (2) While the best μ values vary from dataset to dataset, we can observe the trade-off explained in Section 3.1.2 on all datasets: a larger μ value leads to higher quality but less relations. It is also expected that the best μ value increases with the increase of the sizes of training datasets (i.e., $\mu = 10$ for Tmall, $\mu = 30$ for RetailRocket, and $\mu = 70$ for Last.fm). (3) In general, MGIR's performance drops as δ increases because a larger δ value will filter out more useful incompatible relations. We can also observe that even when δ is relatively large, MGIR can still achieve good performance compared with MGIR w/o In (i.e., $\delta = 1$). This fact again demonstrates the importance of considering incompatible relations (i.e., even a small number of incompatible relations can substantially boost performance).

4.5 Impact of Fusion Methods (RQ4)

In Section 3.2, we discussed different ways of fusing base embeddings and different global embeddings (i.e., Eq. (6), (8) and (10)). We conduct a series of experiments to justify our design choices. Here we consider several widely used choices: mean pooling, sum pooling, convolution and MLP. The experimental results are presented in Table 3. Due to the space limitation, we only report the results on Tmall. The observations are similar on RetailRocket and Last.fm. It can be seen that adopting convolution operations as the fusion strategy outperforms other methods for learning sequential and co-occurrence global item embeddings, while using mean pooling as the fusion method achieves the best performance for

Table 3: Impact of different fusion methods on Tmall.

	Sequential		Co-occurrence		Incompatible	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
Mean	35.81	16.77	35.67	16.80	**36.41**	**17.42**
Sum	35.40	16.55	35.59	16.64	35.93	17.08
Conv	**36.41**	**17.42**	**36.41**	**17.42**	35.91	17.12
MLP	36.19	17.21	35.87	17.00	35.91	17.12

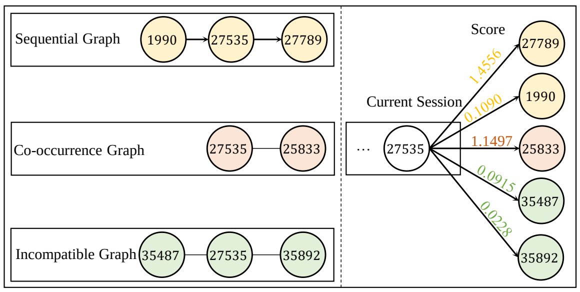


Figure 6: A case study to show how different global relations affect next-item recommendation.

learning incompatible global item embeddings. The results confirm the correctness of our previous theoretical analysis.

4.6 Case Study

Finally, we conduct a case study to illustrate how the multi-faceted global item relations captured in MGIR help make more accurate recommendations. In Figure 6, we show a session whose last interacted item is of ID 27535 from Tmall. From the global sequential graph, we can learn that item ID 27789 is usually purchased after purchasing item ID 27535, while ID 1990 is usually purchased before item ID 27535. Intuitively, we should recommend item ID 27789 instead of item ID 1990. This is well captured by MGIR, which gives a much higher score to recommend item ID 27789. From the global co-occurrence graph, we can observe that item ID 27535 and item ID 25833 are usually purchased together without a particular order. Thus MGIR yields a high score for item ID 25833. From the incompatible graph, we can derive that having item ID 27535 in the current session should make item ID 35487 and item ID 35892 less likely as the next item. Similarly, this is well aligned with the scores generated by MGIR. As such, we deem that MGIR is able to well leverage the three types of global item relations to generate more accurate recommendations.

5 CONCLUSION

In this paper, we studied the problem of session-based recommendation from a new perspective—how to learn multi-faceted session-independent global item relations to enhance session representations. We identified three types of global item relations, including negative relations that have not been touched upon in the literature. We consequently devised different graph construction methods and a novel MGIR model to effectively make use of the global item relations for improving session-based recommendation. We conducted comprehensive experiments to show that our solution can achieve consistently better performance than a large number of state-of-the-art competitors on multiple benchmark datasets.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China under Grant No. 2020YFB1710200.

REFERENCES

[1] Rakesh Agrawal, Tomasz Imielinski, and Arun N. Swami. 1993. Mining Association Rules between Sets of Items in Large Databases. In *Proceedings of the International Conference on Management of Data (SIGMOD)*. 207–216.

[2] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*. 487–499.

[3] Chen Chen, Jie Guo, and Bin Song. 2021. Dual Attention Transfer in Session-Based Recommendation With Multi-Dimensional Integration. In *Proceedings of the 44th International Conference on Research and Development in Information Retrieval (SIGIR)*. 869–878.

[4] Gaode Chen, Xinghua Zhang, Yanyan Zhao, Cong Xue, and Ji Xiang. 2021. Exploring Periodicity and Interactivity in Multi-Interest Framework for Sequential Recommendation. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*. 1426–1433.

[5] Tianwen Chen and Raymond Chi-Wing Wong. 2020. Handling Information Loss of Graph Neural Networks for Session-Based Recommendation. In *Proceedings of the 26th International Conference on Knowledge Discovery And Data Mining (KDD)*. 1172–1180.

[6] Junsu Cho, Dongmin Hyun, SeongKu Kang, and Hwanjo Yu. 2021. Learning Heterogeneous Temporal Patterns of User Preference for Timely Recommendation. In *Proceedings of the 30th International Conference on World Wide Web (WWW)*. 1274–1283.

[7] Junsu Cho, SeongKu Kang, Dongmin Hyun, and Hwanjo Yu. 2021. Unsupervised Proxy Selection for Session-Based Recommender Systems. In *Proceddings of the 44th International Conference on Research And Development in Information Retrieval (SIGIR)*. 327–336.

[8] Minjin Choi, Jinhong Kim, Joonseok Lee, Hyunjung Shim, and Jongwuk Lee. 2021. Session-Aware Linear Item-Item Models for Session-Based Recommendation. In *Proceedings of the 30th International Conference on World Wide Web (WWW)*. 2186–2197.

[9] Diksha Garg, Priyanka Gupta, Pankaj Malhotra, Lovekesh Vig, and Gautam Shroff. 2019. Sequence and Time Aware Neighborhood for Session-based Recommendations: STAN. In *Proceedings of the 42nd International Conference on Research and Development in Information Retrieval (SIGIR)*. 1069–1072.

[10] Xueliang Guo, Chongyang Shi, and Chuanming Liu. 2020. Intention Modeling from Ordered and Unordered Facets for Sequential Recommendation. In *Proceedings of the 29th International Conference on World Wide Web (WWW)*. 1127–1137.

[11] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-Based Recommendations with Recurrent Neural Networks. In *Proceedings of the 4th International Conference on Learning Representations (ICLR)*.

[12] Cheng Hsu and Cheng-Te Li. 2021. RetaGNN: Relational Temporal Attentive Graph Neural Networks for Holistic Sequential Recommendation. In *Proceedings of the 30th International Conference on World Wide Web (WWW)*. 2968–2979.

[13] Chao Huang, Jiahui Chen, Lianghao Xia, Yong Xu, Peng Dai, Yanqing Chen, Liefeng Bo, Jiashu Zhao, and Jimmy Xiangji Huang. 2021. Graph-Enhanced Multi-Task Learning of Multi-Level Transition Dynamics for Session-Based Recommendation. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence, (AAAI)*. 4123–4130.

[14] Dietmar Jannach and Malte Ludewig. 2017. When Recurrent Neural Networks meet the Neighborhood for Session-Based Recommendation. In *Proceedings of the 11st Conference on Recommender Systems (RecSys)*. 306–310.

[15] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *Proceeding of the 18th International Conference on Data Mining (ICDM)*. 197–206.

[16] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

[17] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural Attentive Session-Based Recommendation. In *Proceedings of the 26th International Conference on Information and Knowledge Management (CIKM)*. 1419–1428.

[18] Chang Liu, Xiaoguang Li, Guohao Cai, Zhenhua Dong, Hong Zhu, and Lifeng Shang. 2021. Noninvasive Self-attention for Side Information Fusion in Sequential Recommendation. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*. 4249–4256.

[19] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 2018. STAMP: Short-Term Attention/Memory Priority Model for Session-Based Recommendation. In *Proceedings of the 24th International Conference on Knowledge Discovery and Data Mining (KDD)*. 1831–1839.

[20] Anjing Luo, Pengpeng Zhao, Yanchi Liu, Fuzhen Zhuang, Deqing Wang, Jiajie Xu, Junhua Fang, and Victor S. Sheng. 2020. Collaborative Self-Attention Network for Session-Based Recommendation. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*. 2591–2597.

[21] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical Gating Networks for Sequential Recommendation. In *Proceedings of the 25th International Conference on Knowledge Discovery and Data Mining (KDD)*. 825–833.

[22] Wenjing Meng, Deqing Yang, and Yanghua Xiao. 2020. Incorporating User Micro-Behaviors and Item Knowledge into Multi-Task Learning for Session-Based Recommendation. In *Proceedings of the 43rd International Conference on Research and Development in Information Retrieval (SIGIR)*. 1091–1100.

[23] Dai Quoc Nguyen, Tu Dinh Nguyen, Dat Quoc Nguyen, and Dinh Q. Phung. 2018. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. 327–333.

[24] Zhiqiang Pan, Fei Cai, Wanyu Chen, Honghui Chen, and Maarten De Rijke. 2020. Star Graph Neural Networks for Session-Based Recommendation. In *Proceedings of the 29th International Conference on Information And Knowledge Management (CIKM)*. 1195–1204.

[25] Ruihong Qiu, Zi Huang, Jingjing Li, and Hongzhi Yin. 2020. Exploiting Cross-session Information for Session-based Recommendation with Graph Neural Networks. *ACM Transactions on Information Systems TOIS* 38, 3 (2020), 22:1–22:23.

[26] Ruihong Qiu, Jingjing Li, Zi Huang, and Hongzhi Yin. 2019. Rethinking the Item Order in Session-Based Recommendation With Graph Neural Networks. In *Proceedings of the 28th International Conference on Information and Knowledge Management (CIKM)*. 579–588.

[27] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing Personalized Markov Chains for Next-Basket Recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*. 811–820.

[28] Jing Song, Hong Shen, Zijing Ou, Junyi Zhang, Teng Xiao, and Shangsong Liang. 2019. ISLF: Interest Shift and Latent Factors Combination Model for Session-Based Recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. 5765–5771.

[29] Yang Sun, Fajie Yuan, Min Yang, Guoao Wei, Zhou Zhao, and Duo Liu. 2020. A Generic Network Compression Framework for Sequential Recommender Systems. In *Proceedings of the 43rd International Conference on Research and Development in Information Retrieval (SIGIR)*. 1299–1308.

[30] Qiaoyu Tan, Jianwei Zhang, Ninghao Liu, Xiao Huang, Hongxia Yang, Jingren Zhou, and Xia Hu. 2021. Dynamic Memory based Attention Network for Sequential Recommendation. In *Proceedings of the 35th AAAI International Conference on Artificial Intelligence (AAAI)*. 4384–4392.

[31] Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved Recurrent Neural Networks for Session-Based Recommendations. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (DLRS@RecSys)*. 17–22.

[32] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the 11st International Conference on Web Search and Data Mining (WSDM)*. 565–573.

[33] Trinh Xuan Tuan and Tu Minh Phuong. 2017. 3D Convolutional Networks for Session-Based Recommendation With Content Features. In *Proceedings of the 11st Conference on Recommender Systems (RecSys)*. 138–146.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*. 5998–6008.

[35] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

[36] Chengkang Wang, Mengting Wan, and Julian J. McAuley. 2018. Recommendation Through Mixtures of Heterogeneous Item Relationships. In *Proceedings of the 27th International Conference on Information And Knowledge Management (CIKM)*. 1143–1152.

[37] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2020. Make It a Chorus: Knowledge- and Time-Aware Item Modeling for Sequential Recommendation. In *Proceedings of the 43rd International Conference on Research and Development in Information Retrieval (SIGIR)*. 109–118.

[38] Jian Wang, Badrul Sarwar, and Neel Sundaresan. 2011. Utilizing related products for post-purchase recommendation in e-commerce. In *Proceedings of the 5th Conference on Recommender Systems (RecSys)*. 329–332.

[39] Meirui Wang, Pengjie Ren, Lei Mei, Zhumin Chen, Jun Ma, and Maarten De Rijke. 2019. A Collaborative Session-Based Recommendation Approach With Parallel Memory Modules. In *Proceedings of the 42rd International Conference on Research and Development in Information Retrieval (SIGIR)*. 345–354.

[40] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *Proceedings of the 25th International Conference on Knowledge Discovery and Data Mining (KDD)*. 950–958.

[41] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xianling Mao, and Minghui Qiu. 2020. Global Context Enhanced Graph Neural Networks for Session-Based Recommendation. In *Proceedings of the 43rd International Conference On Research and Development in Information Retrieval (SIGIR)*. 169–178.

[42] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-Based Recommendation With Graph Neural Networks. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*. 346–353.

[43] Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui. 2021. Self-Supervised Graph Co-Training for Session-based Recommendation. In *Proceedings of the 30th International Conference on Information and Knowledge Management (CIKM)*. 2180–2190.

[44] Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Lizhen Cui, and Xiangliang Zhang. 2021. Self-Supervised Hypergraph Convolutional Networks for Session-Based Recommendation. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*. 4503–4511.

[45] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. 2019. Graph Contextualized Self-Attention Network for Session-Based Recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. 3940–3946.

[46] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Jiajie Xu, Victor S. Sheng, Zhiming Cui, Xiaofang Zhou, and Hui Xiong. 2019. Recurrent Convolutional Neural Network for Sequential Recommendation. In *Proceedings of the 28th International Conference on World Wide Web (WWW)*. 3398–3404.

[47] Rui Ye, Qing Zhang, and Hengliang Luo. 2020. Cross-Session Aware Temporal Convolutional Network for Session-Based Recommendation. In *Proceedings of the 20th International Conference on Data Mining Workshops (ICDMW)*. 220–226.

[48] Wenwen Ye, Shuaiqiang Wang, Xu Chen, Xuepeng Wang, Zheng Qin, and Dawei Yin. 2020. Time Matters: Sequential Recommendation with Complex Temporal Information. In *Proceedings of the 43rd International Conference on Research and Development in Information Retrieval (SIGIR)*. 1459–1468.

[49] Lu Yu, Chuxu Zhang, Shangsong Liang, and Xiangliang Zhang. 2019. Multi-Order Attentive Ranking Model for Sequential Recommendation. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*. 5709–5716.

[50] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the 12nd International Conference on Web Search And Data Mining (WSDM)*. 582–590.

[51] Jiahao Yuan, Zihan Song, Mingyou Sun, Xiaoling Wang, and Wayne Xin Zhao. 2021. Dual Sparse Attention Network for Session-Based Recommendation. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*. 4635–4643.